



Combinatoire algébrique des arbres

Samuele Giraudo

► **To cite this version:**

Samuele Giraudo. Combinatoire algébrique des arbres. Autre [cs.OH]. Université Paris-Est, 2011. Français. <NNT : 2011PEST1023>. <pastel-00674619>

HAL Id: pastel-00674619

<https://pastel.archives-ouvertes.fr/pastel-00674619>

Submitted on 27 Feb 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE DE DOCTORAT

pour l'obtention du grade de

Docteur de l'université Paris-Est

Spécialité Informatique

*au titre de l'École Doctorale de Mathématiques
et des Sciences et Techniques de l'Information et de la Communication*

Présentée et soutenue publiquement par

Samuele Giraudo

le 8 décembre 2011

Combinatoire algébrique des arbres

Devant le jury composé par

Frédéric Chapoton	Rapporteur
Sylvie Corteel	Rapporteur
Florent Hivert	Directeur de thèse
Cyril Nicaud	Examineur
Jean-Christophe Novelli	Directeur de thèse
Nathan Reading	Rapporteur
Gilles Schaeffer	Examineur
Xavier Gérard Viennot	Président du jury

Laboratoire d'informatique Gaspard-Monge
UMR 8049 LIGM

5, bd Descartes, Champs-sur-Marne, 77454 Marne-la-Vallée Cedex 2, France

Remerciements

Je pense que l'écriture des remerciements est un moment où se mêlent joie, émotion et souvenirs. Les bons souvenirs restent tandis que les moins bons se fanent et s'oublient. J'ai rencontré pendant mes années de doctorat de nombreuses personnes intéressantes, passionnantes et enrichissantes. Je leur consacre les lignes qui suivent.

Je commence par remercier très chaleureusement Florent Hivert. J'ai connu Florent à l'université de Rouen lorsque j'étais étudiant en deuxième année de licence. Ses qualités pédagogiques, son application et son implication dans l'enseignement m'ont permis et donné l'envie de m'intéresser à la combinatoire. Sans lui, j'aurais ignoré l'existence de cette fabuleuse branche des mathématiques et de l'informatique. Il a été mon directeur de stage de master et m'a fait cadeau d'un sujet fécond qui m'a permis de faire mes premiers pas dans le monde de la recherche. Je le remercie pour la confiance qu'il m'a accordée, sa générosité, la très grande clarté de ses explications et le temps qu'il m'a consacré. Merci également pour tout le temps passé pour la relecture des mes articles.

J'ai connu Jean-Christophe Novelli en 2009 par l'intermédiaire de Florent lors de l'un de ces fameux séminaires du vendredi de l'équipe de combinatoire de Marne-la-Vallée, alors que je terminais mon stage de master. J'ai été frappé par sa grande rigueur scientifique, sa franchise et sa facilité à déchiffrer ce que je souhaite exprimer alors que rien n'est encore clair pour moi. Jean-Christophe a été mon directeur de thèse « sur le terrain » et je souhaite le remercier très sincèrement pour tout ce qu'il a fait pour moi, pour sa patience et pour toutes les fois où je suis venu m'inviter dans son bureau pour lui poser toute sorte de questions. Je remercie également Jean-Christophe pour la confiance et la grande liberté qu'il m'a accordée, c'est tout ce dont je rêvais.

Je remercie mes trois rapporteurs pour avoir accepté de lire mon mémoire et écrire un rapport. En particulier, je remercie Frédéric Chapoton pour ses suggestions qui ont amélioré le texte de manière significative et pour avoir répondu à diverses questions par courriel à plusieurs reprises, Sylvie Corteel pour son accueil chaleureux lors de mes visites au LIAFA où j'ai pu lui présenter mon travail et Nathan Reading pour son partage d'idées et ses encouragements.

C'est un immense honneur de compter parmi mon jury des personnalités aussi marquantes et importantes en combinatoire que Cyril Nicaud, Gilles Schaeffer et Xavier Viennot. Je les remercie d'avoir accepté de faire partie de mon Jury.

L'université de Marne-la-Vallée est une grande famille. Les échanges — professionnels ou non — entre les membres de l'équipe de combinatoire, du laboratoire et de ses visiteurs y sont fructueux et agréables. Je remercie pour cela François Bergeron, Philippe Biane, Gérard Duchamp, Matthieu Josuat-Vergès, Alain Lascoux, Yvan Le Borgne, Jean-Gabriel Luque, Christophe Reutenauer, Jean-Yves Thibon et Nicolas Thiéry. Je remercie également les (anciens) doctorants que j'ai rencontré : Nicolas Borie, Adrien Boussicault, Jean-Paul Bultel, Hayat Cheballah, Valentin Féray, Robin Langer, Rémi Maurice, Pierre-Loïc Méliot, Viviane Pons, Marc Sage et Vincent Vong. Je remercie plus particulièrement ceux qui ont partagé mon bureau : l'ambiance y était parfaite. Je remercie les personnes qui ont travaillé avec moi pour l'enseignement, et tout particulièrement Claire David, pour sa très grande générosité et pour ses conseils précieux et d'une très grande pertinence, à la fois pour l'enseignement et pour ma thèse. Un grand merci aussi à Teresa Gomez-Diaz pour son aide à la préparation des posters et à Patrice Hérault pour ses multiples interventions informatiques. Je remercie tous ceux qui fréquentent « le troisième » et plus particulièrement Marc Zipstein pour sa constante bonne humeur et l'excellente qualité de sa cuisine. Enfin, je souhaite dire merci aux (anciens) membres administratifs de mon laboratoire et de mon école doctorale : Gabrielle Brossard, Sylvie Cach, Séverine Crosson, Marie-Hélène Duprat, Line Fonfrède et Pascale Souliez. Leur présence m'a simplifié la vie de manière considérable.

Finalement, je remercie les membres de ma famille qui ont toujours été présents pour moi : mon Papa, ma Maman, mes frères Carlo et Davide, et ma petite sœur, Chiara.

Combinatoire algébrique des arbres — Algebraic combinatorics on trees

Résumé

Cette thèse se situe dans le domaine de la combinatoire algébrique et porte sur la construction de plusieurs structures combinatoires et algébriques sur différentes espèces d'arbres.

Après avoir défini un analogue du monoïde plaxique dont les classes d'équivalence sont indexées par les couples d'arbres binaires jumeaux, nous proposons un analogue de la correspondance de Robinson-Schensted dans ce contexte. À partir de ce monoïde, nous construisons une sous-algèbre de Hopf de l'algèbre de Hopf des fonctions quasi-symétriques libres dont les bases sont indexées par les couples d'arbres binaires jumeaux.

Ensuite, nous proposons un foncteur combinatoire de la catégorie des monoïdes vers la catégorie des opérades ensemblistes. En utilisant ce foncteur, nous construisons plusieurs opérades qui mettent en jeu divers objets combinatoires. Par le biais d'une construction qui à une opérade associe une algèbre de Hopf non commutative, nous obtenons à partir de l'une des opérades obtenue par notre construction, une algèbre de Hopf basée sur les forêts ordonnées d'arbres plans enracinés. Nous proposons une réalisation polynomiale de cette dernière.

Finalement, nous établissons certaines propriétés vérifiées par les arbres binaires équilibrés dans le treillis de Tamari. Nous montrons que l'ensemble des arbres binaires équilibrés y est clos par intervalle et que les intervalles d'arbres binaires équilibrés ont la forme d'hypercubes. Dans l'objectif de dénombrer ces intervalles, nous introduisons une nouvelle sorte de grammaires d'arbres, les grammaires synchrones. Celles-ci permettent d'obtenir une équation fonctionnelle de point fixe pour la série génératrice des arbres qu'elles engendrent.

Mots-clés :

combinatoire ; algorithmique ; arbre ; algèbre de Hopf ; opérade ; fonction quasi-symétrique libre ; treillis de Tamari ; correspondance de Robinson-Schensted ; réalisation polynomiale.

Abstract

This thesis comes within the scope of algebraic combinatorics and deals with the construction of several combinatorial and algebraic structures on different tree species.

After defining an analogue of the plactic monoid whose equivalence classes are indexed by pairs of twin binary trees, we propose in this context an analogue of the Robinson-Schensted correspondence. From this monoid, we construct a Hopf subalgebra of the Hopf algebra of free quasi-symmetric functions whose bases are indexed by pairs of twin binary trees.

Then, we propose a combinatorial functor from the category of monoids to the category of set-operads. Using this functor, we construct several operads that involve various combinatorial objects. Through a construction that brings a noncommutative Hopf algebra from an operad, we obtain from one of the operads obtained by our construction, a Hopf algebra based on ordered forests of planar rooted trees. We propose a polynomial realization of the latter.

Finally, we establish some properties satisfied by balanced binary trees in the Tamari lattice. We show that the set of balanced binary trees is closed by interval and that the intervals of balanced binary trees have the shape of hypercubes. To enumerate these intervals, we introduce a new kind of tree grammars, namely the synchronous grammars. They allow to obtain a fixed-point functional equation for the generating series of the generated trees.

Keywords:

combinatorics ; algorithmics ; tree ; Hopf algebra ; operad ; free quasi-symmetric function ; Tamari lattice ; Robinson-Schensted correspondence ; polynomial realization.

Table des matières

Introduction	1
I Préliminaires	11
1 Combinatoire élémentaire	13
1.1 Structures combinatoires et algébriques élémentaires	13
1.1.1 Classes combinatoires	13
1.1.2 Monoïdes	15
1.1.3 Ensembles partiellement ordonnés et treillis	16
1.1.4 Espaces vectoriels combinatoires	18
1.2 Combinatoire des mots et des permutations	20
1.2.1 Mots et opérations sur les mots	20
1.2.2 Permutations, opérations sur les permutations et permutoèdre	21
1.3 Combinatoire des graphes et des arbres	22
1.3.1 Graphes	23
1.3.2 Structures arborescentes	23
1.3.3 Arbres binaires et treillis de Tamari	24
2 Algèbres de Hopf combinatoires	29
2.1 Définitions et propriétés de base	30
2.1.1 Algèbres combinatoires	30
2.1.2 Cogèbres combinatoires	33
2.1.3 Bigèbres et algèbres de Hopf combinatoires	35
2.1.4 Réalisations polynomiales d'algèbres de Hopf	38
2.2 Exemples d'algèbres de Hopf combinatoires	40
2.2.1 L'algèbre de Hopf de mélange et de déconcaténation	40
2.2.2 L'algèbre de Hopf de Connes-Kreimer	42
2.2.3 L'algèbre de Hopf des fonctions quasi-symétriques libres	44
3 Opérades	49
3.1 Définitions et propriétés de base	50
3.1.1 Définitions	50
3.1.2 Opérades libres, présentations par générateurs et relations	54
3.1.3 Algèbres sur une opérade	55
3.2 Exemples d'opérades	56
3.2.1 L'opérade commutative associative	56
3.2.2 L'opérade associative	57

3.2.3	L'opérade de Lie	57
3.2.4	L'opérade dendriforme	58
3.3	Groupes et algèbres de Hopf associés à une opérade	58
3.3.1	Groupe associé à une opérade	59
3.3.2	Algèbres de Hopf associées à une opérade	60
II	Combinatoire des couples d'arbres binaires jumeaux	63
4	Le monoïde de Baxter	65
4.1	Bons monoïdes	66
4.1.1	Définitions de base	66
4.1.2	Quelques exemples de bons monoïdes	68
4.1.3	Opérations sur les bons monoïdes	71
4.2	La congruence de Baxter	72
4.2.1	Permutations de Baxter et couples d'arbres binaires jumeaux	72
4.2.2	La relation d'équivalence de Baxter et ses premières propriétés	73
4.2.3	Liens avec d'autres bons monoïdes connus	75
4.3	Une correspondance à la Robinson-Schensted	77
4.3.1	Un algorithme d'insertion à la Robinson-Schensted	77
4.3.2	Une correspondance à la Robinson-Schensted	79
4.3.3	Extraction de mots	80
4.3.4	Version itérative de l'algorithme d'insertion	86
4.4	Un treillis sur les couples d'arbres binaires jumeaux	88
4.4.1	Congruences de treillis	88
4.4.2	Le treillis des couples d'arbres binaires jumeaux	89
4.4.3	Diagrammes de Tamari doubles	91
5	L'algèbre de Hopf des couples d'arbres binaires jumeaux	95
5.1	Construction de sous-algèbres de Hopf de FQSym	96
5.1.1	La construction	96
5.1.2	L'algèbre de Hopf des tableaux de Young standard	99
5.1.3	L'algèbre de Hopf des arbres binaires	99
5.1.4	L'algèbre de Hopf des fonctions symétriques non commutatives	101
5.2	Quelques propriétés générales des sous-algèbres de Hopf de FQSym	101
5.2.1	Produits et intervalles	102
5.2.2	Bases multiplicatives et générateurs libres	103
5.2.3	Bases booléennes	104
5.2.4	Structure de bigèbre bidendriforme	105
5.3	L'algèbre de Hopf combinatoire des couples d'arbres binaires jumeaux	109
5.3.1	L'algèbre de Hopf Baxter	109
5.3.2	Propriétés de Baxter	110
III	Monoïdes, opérades et algèbres de Hopf combinatoires	121
6	Construction d'opérades à partir de monoïdes	123
6.1	Un foncteur des monoïdes vers les opérades ensemblistes	124
6.1.1	La construction	124
6.1.2	Propriétés de la construction	126
6.2	Quelques opérades issues de la construction	127
6.2.1	Une opérade sur les endofonctions	128
6.2.2	Une opérade sur les fonctions de parking	129

6.2.3	Une opérade sur les mots tassés	130
6.2.4	Une opérade sur les permutations	131
6.2.5	Une opérade sur les arbres plans enracinés	132
6.2.6	Une famille d'opérades sur les k -chemins de Dyck	135
6.2.7	Une opérade sur les arbres de Schröder	137
6.2.8	Une opérade sur les chemins de Motzkin	139
6.2.9	Une opérade sur les compositions d'entiers	141
6.2.10	Une opérade sur les animaux dirigés	142
6.2.11	Une opérade sur les compositions d'entiers segmentées	145
6.2.12	L'opérade diassociative	146
6.3	L'algèbre de Hopf de l'opérade des arbres plans enracinés	147
6.3.1	L'algèbre de Hopf des forêts planes enracinées	147
6.3.2	Une réalisation polynomiale	150
IV	Combinatoire des arbres binaires équilibrés	157
7	Grammaires synchrones	159
7.1	Arbres à bourgeons et grammaires synchrones	160
7.1.1	Arbres à bourgeons	160
7.1.2	Grammaires synchrones	160
7.1.3	Conditions sur les grammaires synchrones	163
7.2	Grammaires synchrones et dénombrement	165
7.2.1	Série génératrice d'une grammaire synchrone	165
7.2.2	Calculer la série génératrice d'une grammaire synchrone	166
7.2.3	Spécialisations et raffinements	167
7.3	Exemples de grammaires synchrones	168
7.3.1	La grammaire synchrone des arbres binaires parfaits	168
7.3.2	La grammaire synchrone des arbres 2, 3-équilibrés	168
7.3.3	La grammaire synchrone des arbres binaires équilibrés	169
8	Arbres binaires équilibrés et treillis de Tamari	171
8.1	Clôture par intervalles des arbres binaires équilibrés	172
8.1.1	Arbres binaires équilibrés	172
8.1.2	Rotations dans un arbre binaire équilibré	173
8.1.3	Un invariant de déséquilibre	174
8.2	Motifs de déséquilibre et arbres binaires équilibrés	181
8.2.1	Occurrences et motifs de déséquilibre	182
8.2.2	Arbres équilibrés maximaux, intérieurs et mixtes	182
8.3	Forme des intervalles d'arbres binaires équilibrés	186
8.3.1	Intervalles d'arbres binaires équilibrés et hypercubes	186
8.3.2	Énumération des intervalles d'arbres binaires équilibrés	188
8.4	Autres familles d'arbres binaires closes par intervalle	191
8.4.1	Arbres binaires équilibrés généralisés	191
8.4.2	Arbres binaires équilibrés en nombre de nœuds	194
8.4.3	Arbres binaires ayant une canopée fixée	197
8.4.4	Arbres binaires ayant un indice de Narayana fixé	198
	Perspectives	201
	Bibliographie	205
	Index	213

Table des figures

1.1	Le diagramme de Hasse du permutoèdre droit d'ordre 4.	22
1.2	Un arbre binaire de taille 8, dessiné avec ses feuilles.	24
1.3	Un arbre binaire dont la canopée est 0100101.	25
1.4	L'opération de rotation dans un arbre binaire. La rotation droite en y envoie l'arbre binaire T_0 sur l'arbre binaire T_1 . La rotation gauche en y envoie quant à elle T_1 sur T_0	25
1.5	Les diagrammes de Hasse des treillis de Tamari d'ordre 3 et 4.	26
1.6	Conditions sur les étiquettes des sous-arbres d'un arbre binaire de recherche. . .	27
2.1	Un arbre enraciné T , et le couple $T _E := (T', F)$ où $E := \{x, y, z, t\}$ est une coupe admissible de Connes-Kreimer.	43
4.1	Un couple d'arbres binaires jumeaux de taille 8.	73
4.2	Un couple d'arbres binaires de recherche jumeaux de taille 10.	73
4.3	La classe d'équivalence de Baxter du mot $u := 2415253$ et de la permutation $2516374 = \text{std}(u)$. Les arêtes représentent les relations d'adjacence de Baxter. . .	74
4.4	Le \mathbb{P} -symbole de Baxter du mot 2415253.	77
4.5	Le \mathbb{Q} -symbole de Baxter du mot 2415253.	79
4.6	Une exécution de l'algorithme EXTRAIREBAXTER sur le couple d'arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation de Baxter calculée est 562134.	83
4.7	Une exécution de l'algorithme EXTRAIREMIN sur le couple d'arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation calculée est 521364 et est le plus petit élément de sa classe de Baxter.	85
4.8	Un arbre binaire de recherche droit T et les arbres binaires $T_{\leq 2}$ et $T_{> 2}$	86
4.9	Les étapes de calcul du \mathbb{P} et du \mathbb{Q} -symbole de Baxter de $u := 2415253$	87
4.10	Le permutoèdre d'ordre 4 découpé en classes d'équivalence de Baxter.	90
4.11	Le diagramme de Tamari 9021043100 dans sa représentation graphique et ses diagonales.	91
4.12	Les diagrammes de Tamari doubles de tailles 0, 1, 2 et 3.	92
4.13	Un intervalle du treillis de Baxter d'ordre 5 dont les sommets sont interprétés comme des couples d'arbres binaires jumeaux et comme des diagrammes de Tamari doubles.	94
5.1	Le graphe gradué $G_{\mathbf{P}}$ associé à Baxter et restreint aux sommets de taille inférieure à quatre.	113
5.2	Le graphe gradué $G_{\mathbf{P}^*}$ associé à Baxter [*] et restreint aux sommets de taille inférieure à quatre.	114

5.3	Le diagramme des sous-algèbres de Hopf connues de FQSym . Les flèches \succrightarrow sont des injections d'algèbres de Hopf.	120
6.1	Le diagramme des sous-opérades ns et quotients de TN . Les flèches \succrightarrow (resp. \rightarrow) sont des morphismes injectifs (resp. surjectifs) d'opérades ns.	128
6.2	Un arbre plan enraciné et l'élément correspondant de l'opérade ns <i>APE</i>	134
6.3	Un exemple de substitution partielle dans l'opérade ns <i>APE</i>	134
6.4	Un 2-chemin de Dyck et l'élément correspondant de l'opérade ns <i>FCat</i> ⁽²⁾	136
6.5	Un exemple de substitution partielle dans l'opérade ns <i>FCat</i> ⁽²⁾	136
6.6	Un arbre de Schröder et l'élément correspondant dans l'opérade ns <i>Schr</i>	139
6.7	Un chemin de Motzkin et l'élément correspondant dans l'opérade ns <i>Motz</i>	140
6.8	Deux exemples de substitutions partielles dans l'opérade ns <i>Comp</i>	142
6.9	Un animal dirigé de taille 21.	143
6.10	Une forêt ordonnée d'arbres plans enracinés et son ensemble d'arêtes.	147
6.11	Un arbre plan enraciné T et le couple $T _I := (S, F)$ où I est le coloriage admissible qui contient les arêtes $r_{20}, r_{32}, r_{53}, r_{63}, r_{72}, r_{80}$ et $r_{10,8}$	149
6.12	Un exemple d'exécution de l'algorithme ÉTIQUETAGEMIN.	151
7.1	Une dérivation dans la grammaire synchrone S_{epi}	162
7.2	Le graphe de génération d'ordre 2 de la grammaire synchrone S_{epi}	162
7.3	Les étapes de dérivation pour la génération d'un arbre binaire équilibré par la grammaire synchrone S_{bal}	170
8.1	Un arbre binaire étiqueté par les mesures de déséquilibre de chacun de ses nœuds.	173
8.2	Les arbres binaires équilibrés de taille inférieure à 6.	173
8.3	Le nœud x est un témoin de déséquilibre de l'arbre binaire T	175
8.4	La suite $(S_{x_1}, \dots, S_{x_\ell})$ des sous-arbres associée au nœud x_1 dans l'arbre binaire T	175
8.5	Quelques exemples de mots des hauteurs dans l'arbre binaire T : $\text{mh}_T(x) = 221$, $\text{mh}_T(y) = 0021$ et $\text{mh}_T(z) = 01$	176
8.6	Une rotation déséquilibrante en r est appliquée à l'arbre binaire équilibré T_0	178
8.7	Le deuxième cas : r est un ancêtre y et r est à droite de y	179
8.8	Le troisième cas : r est à droite de y et n'est pas son ancêtre dans T_1	180
8.9	Les diagrammes de Hasse des douze premiers sous-posets du treillis de Tamari restreint aux arbres binaires équilibrés.	188

Introduction

Avant-propos

La principale activité en combinatoire algébrique, discipline à mi-chemin entre l’informatique et les mathématiques, porte sur l’étude de problèmes classiques de combinatoire comme le dénombrement, l’énumération, la génération aléatoire ou la construction de bijections entre divers objets, mais aussi des problèmes d’analyse d’algorithmes, en s’appuyant sur des méthodes propres à l’algèbre. Réciproquement, l’étude de structures algébriques utilise avec grand profit des moyens informatiques, algorithmiques et des techniques propres à la combinatoire. Ce lien — que l’on peut même qualifier de symbiose car le bénéfice est réciproque — entre algèbre et combinatoire se manifeste par l’émergence de constructions algébriques de diverses sortes dont le point commun réside dans le fait que des structures combinatoires sont mises en jeu. Une très vaste palette de structures algébriques est aujourd’hui connue ; chacune d’entre elles possède ses nuances et ses caractéristiques propres, permet de décrire plus ou moins précisément les contours d’un problème donné et illumine sous un éclairage nouveau certaines propriétés combinatoires. Le formalisme et la rigidité de l’algèbre fournissent gratuitement des questions de base à se poser, et, inversement, des considérations combinatoires peuvent donner lieu à des questions algébriques. Il est ainsi à la fois naturel et digne d’intérêt de construire des structures algébriques sur divers objets combinatoires.

Informatique et arbres

Les *arbres* tiennent le rôle du personnage principal dans notre travail. Ces objets combinatoires furent introduits il y a plus d’un siècle par Cayley [Cay57] et possèdent aujourd’hui un réel don d’ubiquité puisqu’ils apparaissent dans une multitude de contextes d’apparence divers. En informatique, ils sont utilisés en tant que structures de données pour organiser de l’information [AVL62], [Tar87], [AU94], [Knu98], [CLRS03]. Ils permettent également de modéliser des arbres syntaxiques dans le but de représenter ou d’évaluer une expression, notamment en théorie de la compilation [ASU86], [AU94]. De manière similaire, il est d’usage courant de considérer l’arbre des appels récursifs d’un algorithme pour étudier son comportement et sa complexité [CLRS03]. Ils jouent parallèlement un rôle évident en combinatoire [Knu04], par exemple dans le contexte des arbres de génération [Wes90] et en théorie des espèces de structures [Joy81], [BLL94]. Cette théorie, développée par l’école québécoise, offre un cadre combinatoire pour la conception et la manipulation de séries génératrices.

Plusieurs questions de nature algorithmique ou combinatoire sur les arbres restent ouvertes et régulièrement discutées dans la communauté de combinatoire. À titre d’exemple, l’étude des algorithmes introduits par Adelson-Velsky et Landis dans les années 1960 sur les arbres binaires équilibrés [AVL62], [Tar87], [Knu98] n’est toujours pas complète à ce jour. Il n’existe en effet pas d’étude fine sur le comportement de ces algorithmes dans le cas moyen. La construction

de structures algébriques sur les arbres offre un angle d'attaque riche et intéressant pour la résolution, complète ou partielle, de ces problèmes. C'est là précisément le point principal de ce travail, à savoir, l'étude de la combinatoire et de l'algorithmique de diverses espèces d'arbres par le biais de la conception d'outils et de structures algébriques qui les mettent en jeu. Il existe dans ce but principalement deux structures algébriques qui semblent prometteuses pour l'algébrisation d'objets combinatoires, et en particulier des arbres.

Algèbre et arbres

L'une d'entre elles, et celle qui renferme en un sens le plus d'information est la structure d'algèbre de Hopf. Ces structures, historiquement introduites dans le contexte de la topologie algébrique pour l'étude des groupes de Lie, se révèlent centrales en combinatoire algébrique [JR79]. Des références classiques sur le sujet sont [Swe69], [Abe80] et [Car06]. La très grande majorité des objets combinatoires sont constitués de briques de base, que l'on peut qualifier de générateurs. Il est de ce fait raisonnable de vouloir construire un objet — par exemple pour mettre en évidence certaines de ses caractéristiques — en *assemblant* des briques de base ou d'autres objets entre eux. Appelons m le processus qui à deux objets associe ceux que l'on peut construire à partir de l'assemblage de ces derniers. D'un point de vue complémentaire, il est tout aussi raisonnable de *désassembler* un objet en plusieurs briques de base ou en deux objets plus petits. Appelons Δ le processus qui à un objet associe toutes les manières de le décomposer en deux parties. Lorsque m , appelé *produit*, et Δ , appelé *coproduct*, sont des applications linéaires d'espaces vectoriels et lorsqu'elles vérifient un certain nombre de relations raisonnables de compatibilité, on obtient une *algèbre de Hopf*.

L'autre structure algébrique qui encode un tout autre type d'information est celle d'opérade. Celles-ci, indépendamment de leur découverte dans le cadre de la topologie algébrique et l'étude des espaces de lacets par May [May72] et par Boardman et Vogt [BV73], possèdent une place de choix en combinatoire algébrique. Le leitmotiv de la notion d'opérade, autrement que d'assembler des objets entre eux comme dans le cas des algèbres de Hopf, est de *substituer* des objets à des endroits particuliers dans d'autres. L'observation principale qui motive cette idée est que l'on peut voir un objet combinatoire comme un agencement qui possède des *secteurs de substitution*, qui, lorsque l'on place d'autres objets dans ces secteurs, donne un objet plus gros. Notons la ressemblance avec les grammaires formelles [Cho56], [Aut94], [HMU00] qui permettent d'engendrer des langages par substitutions successives de mots. Pour prendre un exemple simple, on peut construire, à partir de deux arbres plans enracinés S et T , un arbre plan enraciné $S \circ_i T$ obtenu en substituant la i^{e} feuille de S en partant de la gauche par la racine de T . L'opérateur \circ_i , appelé *opérateur de substitution partielle*, permet de substituer un objet combinatoire sur le i^{e} secteur de substitution d'un autre. Lorsque \circ_i est défini sur un espace vectoriel, qu'il vérifie un certain nombre de relations, notamment des relations d'associativité et que l'espace vectoriel est muni d'une action du groupe symétrique vérifiant aussi des relations, on obtient une opérade basée sur des objets combinatoires.

Une autre manière de regarder les opérades, et c'est d'ailleurs le point de vue le plus fréquemment adopté puisque c'est là le principal point fort de cette théorie, est de considérer ses éléments comme des opérateurs abstraits que l'on peut composer — ou substituer — entre eux pour en former de plus gros. Les opérades contiennent ainsi des opérateurs de base et de nouveaux peuvent être créés en les assemblant. Ceux-ci peuvent être vus comme des arbres syntaxiques, soumis ou non à certaines relations. Le point fort de cette théorie est que bon nombre de structures algébriques sont gouvernées par une opérade particulière et qu'elles peuvent être comparées par l'intermédiaire de morphismes d'opérade. Par exemple, les algèbres associatives sont gouvernées par l'opérade associative, les algèbres de Lie par l'opérade *Lie*, et il existe un morphisme injectif de l'opérade *Lie* vers l'opérade associative qui permet de retrouver le fait bien connu qu'à partir d'un produit m d'une algèbre associative, un crochet de Lie $[-, -]$ peut en être déduit en posant $[x, y] := xmy - ymx$. Quelques références à la fois générales et intro-

ductives sur les opérades sont [GK94], [Mar06], ainsi que [LV10], mais aussi [Zin10] qui présente une grande gamme de type d'algèbres, d'opérades, et de liens entre elles.

Monoïdes et structures ordonnées

De manière complémentaire aux algèbres de Hopf et aux opérades qui sont des structures relativement riches et contraintes, deux structures algébriques beaucoup plus pauvres sont omniprésentes en combinatoire : les monoïdes et les ensembles partiellement ordonnés. Le concept de monoïde est très minimaliste, il dispose en effet seulement d'un produit associatif qui admet un élément neutre, mais son utilité dans divers champs de la combinatoire n'est plus à démontrer. Citons à ce propos le célèbre monoïde plaxique [LS81], [Lot02] qui est un monoïde sur les tableaux de Young. Son introduction est le premier pas vers une démonstration de la règle de Littlewood-Richardson [LR34], [DHT02], un algorithme permettant de calculer le produit de deux fonctions de Schur (voir par exemple [Mac95]). De manière similaire, les ordres partiels et les treillis apparaissent en combinatoire, et la théorie des fonctions de Möbius intervient dans des problèmes de dénombrement et dans le principe d'inclusion-exclusion [Sta11]. Ils sont également fréquemment utilisés dans la construction de nouvelles bases, par exemple d'algèbres de Hopf ou d'opérades [HNT05], [AL07].

Constructions algébriques et combinatoires

Toutes ces structures algébriques n'apparaissent la plupart du temps pas de manière isolée les unes par rapport aux autres. Bien souvent, elles s'agencent dans des diagrammes dont les flèches sont des injections ou des surjections. Ces morphismes, souvent calculables par l'intermédiaire d'algorithmes combinatoires qui peuvent tout à fait être implantés informatiquement, mettent en évidence certains liens, évidents ou plus subtils, entre différents objets combinatoires. Il existe en outre des constructions qui, à partir de la donnée d'une structure algébrique, en produisent d'autres. Nous considérons dans ce travail plusieurs de ces constructions.

L'un des principaux points forts de la théorie des opérades et des algèbres de Hopf réside pour nous dans leur contribution à la combinatoire énumérative. En effet, les descriptions apportées par ces structures des objets combinatoires en tant qu'assemblages particuliers d'objets plus petits mènent à la construction de séries génératrices. Même si, à ce jour, la plupart des séries génératrices ainsi obtenues ne sont pas nouvelles, ce moyen algébrique de les obtenir offre un point de vue intéressant en énumération.

Contexte de ce travail

Ce travail de recherche se décline en trois volets indépendants mais qui reposent cependant sur une même toile de fond : celle de construction de structures algébriques sur des objets combinatoires, en particulier sur diverses espèces d'arbres. L'objectif premier étant de construire des outils algébriques pour fournir un angle d'attaque pour la résolution de problèmes combinatoires et algorithmiques.

Algèbres de Hopf combinatoires

La notion heuristique d'*algèbre de Hopf combinatoire* capture les algèbres de Hopf graduées et dont les bases sont indexées par des objets combinatoires. L'algèbre de Hopf des fonctions symétriques Sym est le prototype même de cette notion. Ses éléments sont des fonctions symétriques [Mac95] et ses bases sont indexées par les partitions d'entiers. En ces termes, et pour certaines bases de Sym , le produit et le coproduit se calculent par l'intermédiaire d'opérations élémentaires sur ces objets. Dans ces dernières années, un grand nombre d'algèbres de Hopf combinatoires furent introduites et étudiées — citons par exemple [GKL⁺95], [MR95], [LR98],

[CK98], [HNT02], [NT04], [NTT04], [NT07] et [HNT08a]. Celles-ci généralisent **Sym** au sens où cette dernière se retrouve en tant que sous-algèbre ou quotient de Hopf. Divers objets combinatoires comme les mots, les permutations, diverses familles d'arbres, de forêts et des graphes sont ainsi munis d'une telle structure.

Force est de constater que la grande majorité des produits et des coproduits dans les algèbres de Hopf combinatoires se calculent selon des algorithmes combinatoires dont la complexité, néanmoins variable, est bien palpable. Ceci contraste, d'un autre côté, avec le produit des polynômes non commutatifs, qui lui, est extrêmement simple. Il est de ce fait parfaitement naturel de chercher à encoder de *manière fidèle* les éléments d'une algèbre de Hopf combinatoire par des polynômes de sorte que l'on puisse déduire le produit de l'algèbre en passant par le produit des polynômes qui représentent ses éléments. De plus, on cherche aussi à pouvoir calculer le coproduit d'un élément de l'algèbre de Hopf en question en développant le polynôme qui l'encode sur un alphabet de variables transformé [KLT97], [Hiv03]. Ce procédé, nommé *réalisation polynomiale d'algèbres de Hopf*, a le mérite de simplifier considérablement la théorie. Bon nombre d'algèbres de Hopf possèdent ainsi leur propre réalisation polynomiale [DHT02], [HNT02], [NT07], [FNT10]. De ce fait, la théorie des fonctions symétriques associe des polynômes symétriques à des partitions d'entiers, la théorie des réalisations polynomiales d'algèbres de Hopf associe quant à elle des polynômes généralisant ces derniers à divers autres objets combinatoires.

L'une des algèbres de Hopf combinatoires qui occupe une place de premier plan dans notre travail est l'algèbre de Hopf définie par Malvenuto et Reutenauer [MR95]. Celle-ci est basée sur les permutations et peut être considérée comme une généralisation de **Sym** puisque cette dernière en est un quotient. Son produit s'exprime par un mélange de permutations et son coproduit, par des opérations appelées déconcaténation et standardisation. Cette structure, très riche tant du point de vue combinatoire que du point de vue algébrique, fait intervenir le permutoèdre, un treillis sur les permutations dont les relations de couverture sont les transpositions élémentaires qui augmentent les longueurs des permutations. Cette algèbre de Hopf fit l'objet de divers travaux [Mal93], [DHT02], [AS05], [DHNT08]. Les polynômes réalisateurs de l'algèbre de Hopf de Malvenuto-Reutenauer, construits en [DHT02], forment l'espace des *fonctions quasi-symétriques libres*. Munie de cette réalisation polynomiale, cette algèbre de Hopf fut rebaptisée **FQSym**.

Comme nous venons de le mentionner, **FQSym** généralise l'algèbre de Hopf des fonctions symétriques **Sym**. Il existe cependant de nombreuses algèbres de Hopf combinatoires situées *entre* ces deux dernières — au sens où elles s'injectent dans **FQSym** et **Sym** s'y injecte ou s'y retrouve en tant que quotient. Citons par exemple l'algèbre de Hopf **FSym** [PR95] basée sur les tableaux de Young standard, dont la réalisation polynomiale constitue l'algèbre de Hopf des *fonctions symétriques libres* [DHT02], l'algèbre de Hopf **Bell** [Rey07] basée sur les partitions d'ensembles, l'algèbre de Hopf **PBT** de Loday-Ronco [LR98], [HNT05] basée sur les arbres binaires, ainsi que l'algèbre de Hopf **Sym** [GKL⁺95] basée sur les compositions d'entiers, dont la réalisation polynomiale constitue l'algèbre des *fonctions symétriques non commutatives*.

Il est remarquable que toutes ces structures découlent de la définition d'une règle de réécriture sur les mots qui mène à la construction d'une congruence, puis d'un quotient du monoïde libre, et dont les classes d'équivalence sont indexées par des objets combinatoires. Par exemple, **FSym** est construite à partir du monoïde plaxique, qui lui-même est obtenu comme un quotient du monoïde libre par les relations de Knuth [Knu70]. Les classes d'équivalence de mots induites par les relations de Knuth sont indexées par les tableaux de Young. Dans un contexte algorithmique, Schensted [Sch61] conçut un algorithme pour répondre au problème de calculer en temps $n \log(n)$ la longueur du plus long sous-mot croissant d'un mot de longueur n donné sur un alphabet totalement ordonné. L'algorithme de Schensted produit un tableau de Young sur l'entrée d'un tel mot que l'on insère itérativement lettre à lettre. La réponse de l'algorithme consiste en le nombre de cases contenues dans la première ligne du tableau. Knuth [Knu70] observa une propriété remarquable : deux mots sont congrus par les relations de Knuth si et seulement s'ils produisent le même tableau de Young à l'issue de leur insertion selon l'algorithme de Schensted. L'algorithme

de Schensted offre donc une définition alternative du monoïde plaxique. La correspondance de Robinson-Schensted vise en outre à établir une bijection entre l'ensemble des mots et les couples de tableaux de Young de même forme, dont l'un est standard. Celle-ci est éclairante à plusieurs titres, et offre en particulier une démonstration combinatoire simple du fait que les involutions et les tableaux de Young standard sont en bijection [Knu98].

Plusieurs autres monoïdes qui jouent un rôle *analogue* au monoïde plaxique furent récemment construits. L'un d'entre eux, le monoïde sylvestre [HNT02], [HNT05] est le point de départ d'une construction de l'algèbre de Hopf **PBT** de Loday-Ronco des arbres binaires. L'analogue dans ce contexte de l'algorithme d'insertion de Schensted est l'algorithme d'insertion dans un arbre binaire de recherche [AU94], [Knu98], [CLRS03] et l'analogue de la correspondance de Robinson-Schensted permet d'établir une bijection entre l'ensemble des mots et des couples d'arbres binaires étiquetés dont l'un est un arbre binaire de recherche et l'autre un arbre binaire décroissant. En outre, **PBT** s'injecte dans **FQSym** de la manière suivante. Les éléments de base \mathbf{P}_T de **PBT** sont indexés par les classes sylvestres de permutations, ou, de manière équivalente, par les arbres binaires non étiquetés. Un élément \mathbf{P}_T est défini comme la somme des éléments \mathbf{F}_σ de **FQSym** tels que l'insertion de la permutation σ suivant l'analogue de l'algorithme de Robinson-Schensted donne un arbre binaire de forme T .

D'autres congruences et monoïdes quotients du monoïde libre furent introduits et étudiés, comme le monoïde hypoplaxique [KT97], [Nov98] et le monoïde de Bell [Rey07], qui proposent chacun d'eux un analogue d'un algorithme d'insertion et d'une correspondance de Robinson-Schensted, ainsi qu'une sous-algèbre de Hopf de **FQSym** basée sur les classes d'équivalence de permutations sous les congruences de ces monoïdes. Ces derniers, tout comme le monoïde sylvestre, parce qu'ils partagent des propriétés similaires au monoïde plaxique, en sont des analogues et sont qualifiés de *bon monoïdes* [Hiv04].

Opérades

Le second volet de notre travail est consacré aux opérades et à la construction d'opérades. Le lien avec le précédent volet tient principalement au fait que nous considérons des constructions qui permettent d'obtenir des algèbres de Hopf combinatoires, mais cette fois-ci à partir d'opérades plutôt que de bons monoïdes. Ces dernières années, un grand nombre d'opérades sur divers objets combinatoires furent construites et étudiées. Certaines portent sur diverses espèces d'arbres [Lod01], [CL01], [BF03], [Cha04], [Liv06], [Cha08], d'autres sur des objets plus exotiques comme les arbustes [Cha10], les plantes et les arbres non croisés [Cha06b], ou les moules [Cha06b], [CHNT08]. Cette dernière est l'une des plus intéressantes puisqu'elle contient plusieurs opérades connues, comme l'opérade dendriforme [Lod01], l'opérade associative ou les opérades sur les plantes et arbres non croisés. Il est remarquable de constater que pour la plupart d'entre elles, la substitution partielle et l'action du groupe symétrique s'expriment par l'intermédiaire d'algorithmes combinatoires qui, une fois de plus, sont parfaitement implantables sur ordinateur.

Les opérades, au même titre que les bons monoïdes, peuvent être considérées comme un point de départ dans la construction d'algèbres de Hopf combinatoires. Il existe pour cela au moins deux procédés qui, à partir d'une opérade, mènent à la définition d'une algèbre de Hopf combinatoire. La première construction prend appui sur une famille d'ensembles partiellement ordonnés associés à une opérade ensembliste, obtenus selon un procédé de Méndez et Yang [MY91]. Sous réserve que cette famille vérifie certaines conditions précises, et en utilisant une construction de Schmitt, on associe à cette dernière son algèbre de Hopf d'incidence [Sch94]. Une autre construction, plus directe, vise à construire dans un premier temps un groupe sur la donnée d'une opérade [KM01], [Cha02], [vdL04], [CL07], [Fra08], [Cha08] [LV10], puis à considérer l'algèbre de Hopf des fonctions sur ce groupe. Les groupes obtenus par ce procédé peuvent être considérés comme des produits dérivés de cette construction mais certains exemples montrent

qu'ils s'appliquent à des problèmes de combinatoire énumérative : par exemple, le groupe associé à l'opérade dendriforme apparaît comme un outil fondamental pour compter le nombre d'intervalles dans le treillis de Tamari [Cha06a]. Un lien remarquable entre ces deux constructions d'algèbres de Hopf fut découvert par Chapoton et Livernet [CL07] : il existe un morphisme surjectif de l'algèbre de Hopf des fonctions sur le groupe vers l'algèbre de Hopf d'incidence obtenues à partir d'une même opérade ensembliste.

Arbres binaires équilibrés et rotations

Le troisième volet de cette thèse porte sur l'étude combinatoire et algébrique des arbres binaires équilibrés. Ceux-ci forment une sous-famille des arbres binaires et il est ainsi raisonnable de se poser la question de savoir s'il existe des structures algébriques qui les mettent en jeu, notamment en cherchant des sous-structures ou des quotients de celles déjà construites sur les arbres binaires. L'objectif à terme est de concevoir un outil algébrique pour obtenir un nouveau regard sur les algorithmes qui utilisent les arbres binaires équilibrés comme structures de données, et qui ne sont pour le moment pas analysés de manière exhaustive.

Un arbre binaire est *équilibré* lorsque pour chacun de ses nœuds, la différence en valeur absolue de longueur entre le plus long chemin connectant son fils gauche à une feuille et le plus long chemin connectant son fils droit à une feuille est inférieure à un. Ces arbres interviennent en algorithmique et notamment en tant que structure de donnée pour la représentation d'ensembles dynamiques [AVL62], [Tar87], [Knu98]. L'opération fondamentale dans ce contexte est la *rotation* qui permet de réarranger les sous-arbres d'un arbre binaire de manière à conserver son équilibre lors de création ou de suppression de nœuds.

La rotation dans les arbres binaires intervient dans un tout autre contexte puisque, si l'on munit l'ensemble des arbres binaires à n nœuds d'une relation d'ordre partielle où un arbre binaire T_0 est inférieur à un arbre binaire T_1 si l'on peut obtenir T_1 en appliquant successivement une ou plusieurs rotations orientées dans T_0 , on obtient un treillis connu sous le nom de *treillis de Tamari* [Tam62], [Sta99a], [Knu04]. La combinatoire de ce treillis est très riche. Le treillis de Tamari joue en effet le même rôle pour l'algèbre de Hopf **PBT** que le permutoèdre joue pour **FQSym** : le produit de **PBT** s'exprime en effet comme un intervalle du treillis de Tamari [LR02].

Implantations informatiques

Les résultats théoriques présentés dans ce mémoire s'appuient fortement sur l'implantation informatique. En effet, notre démarche dans toutes les étapes de notre travail est en très grande partie basée sur l'implantation, l'exploration et l'expérimentation informatique. À ces fins, nous avons utilisé le logiciel Sage [S⁺11] et l'extension Sage-Combinat [SCc11] pour implanter la plus grande majorité des algorithmes présentés dans ce mémoire. Les programmes écrits pendant cette thèse se répartissent principalement en trois catégories :

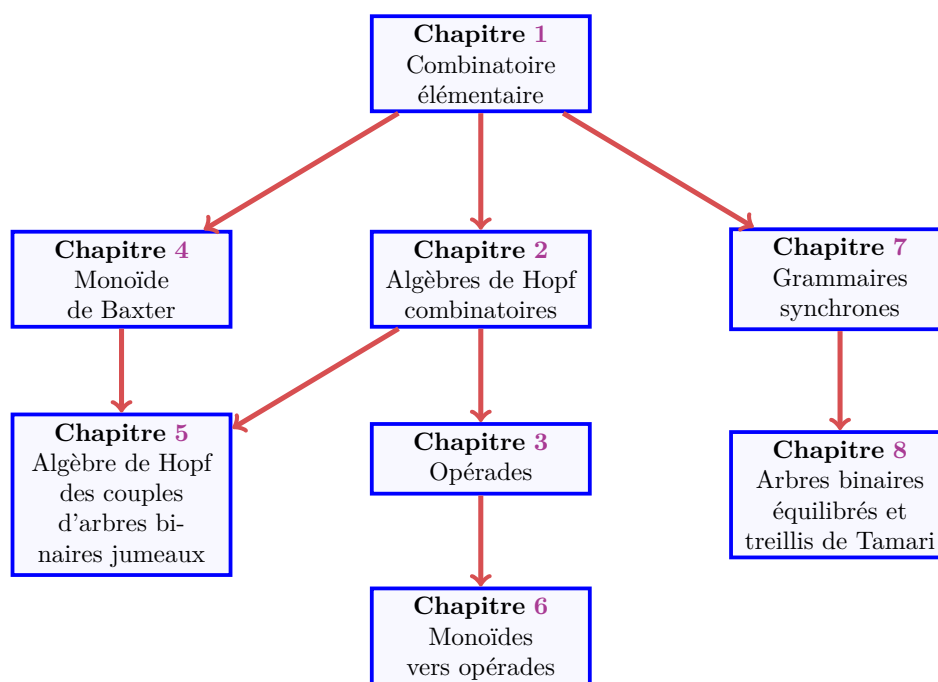
- des algorithmes portant sur des objets combinatoires : mots, permutations, diverses espèces de graphes et d'arbres (environ 11000 lignes de code) ;
- des implantations de plusieurs algèbres de Hopf combinatoires et d'algorithmes en rapport (environ 7000 lignes de code) ;
- des implantations de quelques opérades et d'algorithmes en rapport (environ 2000 lignes de code).

Les volumes ne sont mentionnés qu'à titre indicatif : d'un côté, certains programmes méritent d'être retravaillés et réécrits de manière plus concise, et de l'autre, nous n'avons pas relevé le volume du code que nous avons écrit uniquement pour poser des questions à l'ordinateur — activité qui occupe néanmoins une proportion de plus de 50 % de notre temps. Ces expériences

n'ont en effet aucune valeur de réutilisabilité. Notre code, bien qu'encore expérimental, vise à terme à être introduit dans Sage-Combinat dans le but d'être utilisé, modifié et amélioré librement par qui le souhaite.

Résultats et plan du mémoire

Nos résultats théoriques portent sur trois points correspondant aux parties [II](#), [III](#) et [IV](#) de ce mémoire : la combinatoire des couples d'arbres binaires jumeaux, la construction d'opérades à partir de monoïdes et la combinatoire des arbres binaires équilibrés. Plus précisément les chapitres s'articulent selon le diagramme de dépendances ci-dessous.



Rappels de combinatoire et d'algèbre

Les chapitres [1](#), [2](#) et [3](#) rappellent les définitions fondamentales pour la lecture de ce mémoire. En particulier, les définitions et notations à propos des classes combinatoires, des monoïdes, des ensembles partiellement ordonnés, des treillis, des espaces vectoriels combinatoires, des mots, des permutations, des arbres, des algèbres de Hopf et des opérades sont passées en revue. Nous rappelons également la définition du permutoèdre, du treillis de Tamari, des arbres binaires de recherche, de l'algèbre de Hopf **FQSym** ainsi que certaines de ses propriétés et de la construction qui à une opérade ensembliste associe un groupe et une algèbre de Hopf.

Combinatoire des couples d'arbres binaires jumeaux

L'un de nos premiers objectifs était de compléter le diagramme des sous-algèbres de Hopf de **FQSym** par l'introduction d'une algèbre de Hopf combinatoire inédite — si possible basée sur une espèce d'arbres — et de construire les structures combinatoires associées. Le premier pas consistait ainsi, comme nous l'avons mentionné plus haut, en la définition d'un bon monoïde. Après quelques brèves explorations informatiques, un monoïde dont les classes d'équivalence de permutations semblent être comptées par les permutations de Baxter s'en dégage. Les permutations de Baxter [[Bax64](#)] sont des permutations qui évitent certains motifs [[BS00](#)]. Celles-ci sont

en bijection avec divers objets combinatoires, dont les couples d'arbres binaires jumeaux [DG94], qui sont des couples d'arbres binaires avec une contrainte sur l'orientation de leurs feuilles. Le chapitre 4 est consacré à l'étude de ce monoïde, que nous baptisons *monoïde de Baxter*. Nous construisons un analogue de l'algorithme d'insertion de Robinson-Schensted pour ce monoïde. Celui-ci permet de produire itérativement, sur l'entrée d'un mot, un couple d'arbres binaires jumeaux étiqueté et soumis à diverses conditions. Nous établissons aussi dans ce contexte un analogue de la correspondance de Robinson-Schensted. Celui-ci associe à un mot un couple de couples d'arbres binaires jumeaux étiquetés et fait ressortir la combinatoire des arbres binaires croissants et décroissants. Nous proposons également trois algorithmes qui permettent d'extraire, sur la donnée d'un couple d'arbres binaires jumeaux étiqueté, des mots particuliers dans la classe d'équivalence qu'il encode. De ces algorithmes, nous montrons que les classes du monoïde de Baxter de permutations sont bien en bijection avec les permutations de Baxter. De plus, en montrant que la congruence de Baxter — à l'origine de la définition du monoïde de Baxter — est aussi une congruence du permutoèdre, nous obtenons un treillis quotient du permutoèdre qui est, à bien des points de vue, semblable au treillis de Tamari.

Parallèlement, dans le contexte des congruences de treillis [CS98], [Rea04], Reading [Rea05] construisit une congruence du permutoèdre dont les classes d'équivalence sont indexées par des permutations qui évitent certains motifs. Reading baptisa ces permutations de *permutations de Baxter tordues* et montra qu'il y en a exactement une par classe d'équivalence. De plus, grâce à sa théorie développée dans [Rea05], il obtint gratuitement une sous-algèbre de Hopf de **FQSym** basée sur les permutations de Baxter tordues et un treillis quotient du permutoèdre. Reading observa expérimentalement que les permutations de Baxter tordues de taille n sont en bijection avec les permutations de Baxter de taille n pour n assez grand. Très récemment, Law et Reading [LR10] ont signalé que la première démonstration de cette propriété figure dans un texte non publié de West et donnent eux aussi une démonstration. De manière surprenante, il s'avère dans notre contexte que les permutations de Baxter tordues sont les plus petits éléments de nos classes de Baxter de permutations. En utilisant nos algorithmes d'extraction de mots à partir d'un couple d'arbres binaires jumeaux étiqueté, nous proposons un nouveau moyen de calculer cette bijection. L'algèbre de Hopf de Reading définie en [Rea05] peut ainsi être considérée comme une algèbre de Hopf sur les couples d'arbres binaires jumeaux. Law et Reading [LR10] ont détaillé leur construction et étudié quelques-unes des propriétés de cette algèbre de Hopf.

De notre côté, simultanément et indépendamment du travail de Law et Reading, nous avons étudié, dans le chapitre 5, l'algèbre de Hopf obtenue à partir du monoïde de Baxter et dans le contexte de la théorie des bons monoïdes. Celle-ci, que nous appelons **Baxter**, est indexée par les couples d'arbres binaires jumeaux, est non commutative et non cocommutative. Si l'algèbre de Hopf que nous construisons ainsi est la même que celle que Reading obtint dans [Rea05] dans la théorie des congruences de treillis, le cheminement, la construction et les résultats obtenus sont différents. Nous établissons par exemple une réalisation polynomiale de **Baxter**, montrons qu'elle est libre, qu'elle admet une structure de bigèbre bidendriforme [Foi07], [Foi09], qu'elle est autoduale et dégageons deux bases multiplicatives. Dans ce même chapitre, nous étudions dans un cadre plus général les relations entre les règles de réécriture qui interviennent dans la définition des bons monoïdes et les propriétés algébriques des algèbres de Hopf qui leur sont associées. Nous donnons des conditions suffisantes sur les règles de réécriture pour que l'algèbre de Hopf associée soit libre, autoduale et admette une structure de bigèbre bidendriforme. Nous donnons également une condition suffisante pour que le produit de l'algèbre s'interprète comme un intervalle d'un certain treillis et donnons un moyen de construire des bases multiplicatives. Les résultats contenus dans les chapitres 4 et 5 ont donné lieu à un article [Gir11].

Construction d'opérades à partir de monoïdes

Après avoir construit ces structures combinatoires et algébriques sur les couples d'arbres binaires jumeaux en passant par la théorie des bons monoïdes, et toujours dans l'optique de

construire des algèbres de Hopf combinatoires sur des arbres, il nous a semblé naturel d'explorer une nouvelle approche. Cette approche, comme nous l'avons mentionné plus haut, se base sur la théorie des opérades.

Au lieu d'appliquer les constructions déjà connues qui associent des algèbres de Hopf à une opérade, nous avons souhaité construire des opérades inédites, soit sur des objets combinatoires qui admettent déjà une telle structure, soit sur des objets qui ne rentrent dans aucune opérade connue. Nous sommes ainsi arrivés à la définition d'une construction — que l'on a baptisé *transformation T* — qui à tout monoïde, associe une opérade ensembliste. L'étude de cette construction est l'objet du chapitre 6. Nous montrons que la transformation *T* est un foncteur de la catégorie des monoïdes avec morphismes de monoïde vers la catégorie des opérades avec morphismes d'opérade, et qui respecte de plus les injections et les surjections.

Ainsi, nous utilisons notre construction pour produire diverses opérades et en retrouver des anciennes. En particulier, nous montrons que la transformation *T* appliquée au monoïde additif des entiers naturels contient plusieurs sous-opérades, symétriques ou non, et qui mettent en jeu divers objets combinatoires : les endofonctions, les fonctions de parking, les mots tassés, les permutations, les k -chemins de Dyck, les arbres de Schröder, les chemins de Motzkin, les arbres plans enracinés. Nous obtenons aussi une opérade sur les animaux dirigés et les compositions d'entiers segmentées en tant que sous-opérades de l'opérade obtenue en appliquant la transformation *T* au monoïde additif des entiers modulo 3 et une opérade sur les compositions d'entiers en tant que sous-opérade de l'opérade obtenue en appliquant la transformation *T* au monoïde additif des entiers modulo 2. Nous obtenons également une nouvelle construction de l'opérade diassociative [Lod01] à partir d'une sous-opérade de l'opérade obtenue en appliquant la transformation *T* au monoïde constitué de l'ensemble $\{0, 1\}$ muni de la multiplication. La démarche utilisée ici est expérimentale : nous avons implanté en Sage la transformation *T* ainsi que divers algorithmes de base. Sur la donnée d'un monoïde, l'opérade associée est instanciée et nous pouvons par exemple calculer la série de Hilbert d'une sous-opérade engendrée par un ensemble fini de générateurs.

Ceci étant fait, nous considérons dans la fin du chapitre 6 un analogue non commutatif de la construction qui à une opérade associe un groupe puis une algèbre de Hopf des fonctions sur le groupe. Cette construction est appliquée à l'opérade des arbres plans enracinés obtenue selon notre procédé et donne lieu à une algèbre de Hopf combinatoire non commutative, non cocommutative, dont les bases sont indexées par les forêts ordonnées d'arbres plans enracinés et dont le degré des éléments est leur nombre d'arêtes. Cette algèbre de Hopf ressemble à l'algèbre de Hopf de Connes-Kreimer [CK98], notamment pour l'expression de son coproduit. Nous proposons également une réalisation polynomiale de notre algèbre de Hopf en utilisant des mécanismes similaires à ceux introduits par Foissy, Novelli et Thibon [FNT10] pour établir celle de l'algèbre de Hopf de Connes-Kreimer.

Combinatoire des arbres binaires équilibrés

Maintenant que l'on dispose d'au moins deux procédés différents pour construire des algèbres de Hopf, nous avons souhaité construire une algèbre de Hopf sur les arbres binaires équilibrés. La première approche consiste à chercher des quotients ou des sous-algèbres de Hopf de **PBT** dont les bases sont indexées par les arbres binaires équilibrés. En effet, comme l'a remarqué expérimentalement Hivert, l'ensemble des arbres binaires équilibrés est clos par intervalle dans le treillis de Tamari. Ce treillis étant un objet fondamental dans l'étude de **PBT**, cette observation suggère l'existence d'une sous-algèbre de **PBT** sur les arbres binaires équilibrés. Le début du chapitre 8 est de ce fait consacré à démontrer l'observation d'Hivert. L'ensemble des arbres binaires équilibrés possède d'autres propriétés intéressantes dans le treillis de Tamari. Nous montrons par exemple que les intervalles d'arbres binaires équilibrés ont la forme d'hypercubes. Ainsi, le diagramme de Hasse du treillis de Tamari restreint aux arbres binaires équilibrés est

composé d'hypercubes de différentes dimensions qui se collent entre eux. Nous étudions aussi dans ce chapitre d'autres ensembles d'arbres binaires qui sont clos par intervalle dans le treillis de Tamari, mais aucun ne semble avoir autant de propriétés que l'ensemble des arbres binaires équilibrés. Malgré le fait que les arbres binaires équilibrés jouent un rôle particulier dans le treillis de Tamari, nous ne sommes pour le moment pas parvenus à construire une algèbre de Hopf sur ces objets.

Dans un objectif de dénombrement, nous avons été amenés à considérer un nouvel outil : les *grammaires synchrones*. Comme nous le montrons que le chapitre 7, celui-ci permet d'obtenir les équations fonctionnelles de point fixe pour des séries génératrices dénombrant les arbres binaires équilibrés maximaux dans le treillis de Tamari, ainsi que les intervalles d'arbres binaires équilibrés. Une grammaire synchrone est un type particulier de grammaire d'arbres (voir [CDG⁺07]) qui permet d'engendrer des structures arborescentes en les faisant pousser de la racine jusqu'aux feuilles de manière *synchrone*, c'est-à-dire où une étape de dérivation vise à substituer *la totalité* des symboles non terminaux selon des règles de substitution, exactement comme les branches d'un arbre poussent en parallèle. Sous certaines conditions sur la grammaire synchrone, une équation fonctionnelle de point fixe pour la série génératrice des éléments qu'elle engendre peut en être extraite. À l'aide du formalisme offert par ces grammaires, nous retrouvons par exemple les équations fonctionnelles de point fixe vérifiées par les séries génératrices des arbres binaires équilibrés [BLL88], [BLL94], [Knu98] et des arbres 2,3-équilibrés [Odl82], [FS09]. Les résultats contenus dans les chapitres 7 et 8 ont donné lieu à un article [Gir10].

Première partie

Préliminaires

Chapitre 1

Combinatoire élémentaire

Ce chapitre, premier d'une série de trois chapitres préliminaires, pose les définitions et les concepts de base d'algèbre et de combinatoire utilisés dans ce travail.

Dans le paragraphe 1.1, nous fixons nos notations et rappelons les définitions ainsi que les notions fondamentales sous-jacentes à cinq structures algébriques élémentaires : les classes combinatoires, les monoïdes, les ensembles partiellement ordonnés, les treillis et les espaces vectoriels combinatoires. Les mots sont des objets combinatoires à la fois très simples et très riches. Nous leur consacrons le paragraphe 1.2, où les permutations, qui sont des mots particuliers, sont également passées en revue. Enfin, dans le paragraphe 1.3, les définitions de base à propos des graphes et des arbres sont posées. Nous nous attardons plus particulièrement sur les arbres binaires, qui sont l'un des personnages principaux de ce mémoire.

1.1 Structures combinatoires et algébriques élémentaires

1.1.1 Classes combinatoires

Définition 1.1.1. Une classe combinatoire est un ensemble C muni d'une application taille $|\cdot| \rightarrow \mathbb{N}$ telle que pour tout $n \in \mathbb{N}$, l'ensemble $\{x \in C : |x| = n\}$ est fini.

Un *objet combinatoire* est un élément d'une classe combinatoire, et sa *taille* est son image par l'application $|\cdot|$. On note C_n l'ensemble des objets combinatoires de C de taille n .

À toute classe combinatoire C est associée sa *série génératrice ordinaire* $\mathcal{S}_C(t)$, qui est une série à coefficients entiers positifs, définie par

$$\mathcal{S}_C(t) := \sum_{n \geq 0} \#C_n t^n, \quad (1.1.1)$$

où la notation $\#E$ désigne le cardinal de l'ensemble E . Dans certains cas, il est plus adapté de considérer la *série génératrice exponentielle* $\mathcal{S}_C^e(t)$, qui est définie par

$$\mathcal{S}_C^e(t) := \sum_{n \geq 0} \#C_n \frac{1}{n!} t^n. \quad (1.1.2)$$

Nous utilisons la notation $[t^n]\mathcal{S}_C(t)$ pour accéder au coefficient de t^n dans la série $\mathcal{S}_C(t)$. Ainsi, $[t^n]\mathcal{S}_C(t) = \#C_n$.

Voici quelques exemples de classes combinatoires :

1. la *classe combinatoire vide* \emptyset ne possède aucun objet. Sa série génératrice vérifie $\mathcal{S}_\emptyset(t) = 0$.

2. La *classe combinatoire élémentaire* \mathcal{E} possède un unique objet de taille 0 que l'on note ϵ . Sa série génératrice vérifie $\mathcal{S}_{\mathcal{E}}(t) = 1$.
3. La *classe combinatoire atomique* \mathcal{Z} possède un unique objet de taille 1 que l'on note \bullet . Sa série génératrice vérifie $\mathcal{S}_{\mathcal{Z}}(t) = t$.
4. Si $(C, |\cdot|_C)$ et $(D, |\cdot|_D)$ sont des classes combinatoires, l'*union* de C et de D , notée $C + D$, est la classe combinatoire qui contient les objets de C et de D , munie de l'application taille $|\cdot|$ vérifiant $|x| := |x|_C$ si $x \in C$, et $|x| := |x|_D$ sinon.
5. De même, le *produit* de C et de D , noté $C \times D$ est la classe combinatoire dont les objets sont des couples (c, d) où $c \in C$ et $d \in D$, munie de l'application taille $|\cdot|$ vérifiant $|x| := |x|_C + |x|_D$.
6. Il existe d'autres opérations bien connues sur les classes combinatoires qui à partir d'anciennes, en produisent de nouvelles. Par exemple, la construction *ensemble* prend en entrée une classe combinatoire C et produit la classe combinatoire dont les éléments sont des ensembles d'éléments de C . De même, les constructions *multi-ensemble* et *suite* produisent, sur l'entrée d'une classe combinatoire C respectivement la classe combinatoire dont les éléments sont des multi-ensembles d'éléments de C et la classe combinatoire dont les éléments sont des suites d'éléments de C . Ces constructions, ainsi que les séries génératrices correspondantes, sont décrites par exemple en [FS09].
7. L'ensemble des entiers naturels \mathbb{N} où la taille d'un objet est sa valeur est une classe combinatoire. De manière alternative, \mathbb{N} peut se définir comme étant la classe combinatoire obtenue par l'opération suite appliquée à \mathcal{Z} . Ses éléments de taille inférieure à cinq sont de la forme

$$\epsilon, \bullet, \bullet\bullet, \bullet\bullet\bullet, \bullet\bullet\bullet\bullet, \bullet\bullet\bullet\bullet\bullet,$$

et sa série génératrice est

$$\mathcal{S}_{\mathbb{N}}(t) = \frac{1}{1-t} = 1 + t + t^2 + t^3 + t^4 + \dots \quad (1.1.3)$$

8. La classe combinatoire des *mots* sur l'alphabet $A := \{a, b\}$, notée A^* , contient les suites finies d'éléments de A (voir le paragraphe 1.2.1). De manière alternative, A^* peut se définir comme étant la classe combinatoire obtenue par l'opération suite appliquée à A — ce dernier est vu comme une classe combinatoire qui possède exactement deux objets de taille 1. Ses éléments de taille inférieure à trois sont

$$\epsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb,$$

et sa série génératrice est

$$\mathcal{S}(t) = \frac{1}{1-2t} = 1 + 2t + 4t^2 + 8t^3 + 16t^4 + 32t^5 + \dots \quad (1.1.4)$$

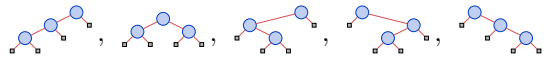
9. La classe combinatoire \mathcal{T} des *arbres binaires* (voir le paragraphe 1.3.3) est définie de manière récursive par

$$\mathcal{T} := \mathcal{E} + \mathcal{T} \times \mathcal{Z} \times \mathcal{T}. \quad (1.1.5)$$

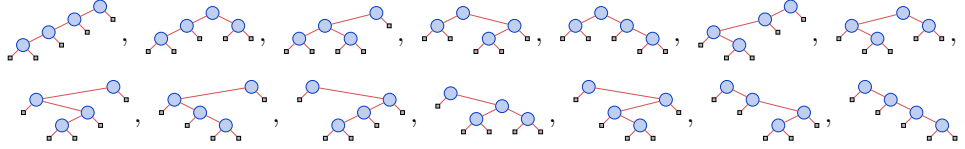
L'expression (1.1.5) spécifie le fait qu'un arbre binaire est soit vide, soit constitué d'un *nœud* et de deux autres arbres binaires. La taille d'un arbre binaire est son nombre de nœuds. Les arbres binaires de taille zéro, un et deux sont



ceux de taille trois,



et ceux de taille quatre,



1.1.2 Monoïdes

Définition 1.1.2. Un monoïde est un ensemble M muni d'une loi de composition interne de la forme $\cdot : M \times M \rightarrow M$ et telle que

- (i) \cdot est associative, i.e., $(x \cdot y) \cdot z = x \cdot (y \cdot z)$ pour tous $x, y, z \in M$;
- (ii) \cdot admet un élément neutre $1 \in M$, i.e., $x \cdot 1 = 1 \cdot x = x$ pour tout $x \in M$.

En fonction des besoins, nous noterons tantôt un monoïde sous la forme de triplet $(M, \cdot, 1)$, tantôt sous la forme de couple (M, \cdot) , ou même tout simplement en faisant référence à son ensemble sous-jacent M si le contexte le permet. Cette dernière convention de notation, où l'on fait référence à l'ensemble sous-jacent d'une structure algébrique pour parler de la structure algébrique elle-même, sera utilisée tout au long de ce mémoire.

Rappelons à présent quelques notions générales sur les monoïdes. Un monoïde $(M, \cdot_M, 1_M)$ est *fini* si l'ensemble M est fini. La partie $M' \subseteq M$ est un *sous-monoïde* de M si $1_M \in M'$ et $x \cdot_M y \in M'$ pour tous $x, y \in M'$. Soit G un sous-ensemble de M . Nous notons par G^* le *monoïde engendré* par G , c'est-à-dire le plus petit sous-monoïde de M qui contient G . Le monoïde M est *finiment engendré* s'il existe un ensemble fini G tel que $M = G^*$. Soit $(N, \cdot_N, 1_N)$ un autre monoïde. Un *morphisme de monoïde* est une application $\theta : M \rightarrow N$ telle que $\theta(1_M) = 1_N$ et pour tous $x, y \in M$,

$$\theta(x \cdot_M y) = \theta(x) \cdot_N \theta(y). \quad (1.1.6)$$

Les deux monoïdes M et N sont *isomorphes* s'il existe un morphisme de monoïde $\theta : M \rightarrow N$ qui est également une bijection. Le monoïde M est *gradué* s'il existe une application $\phi : M \rightarrow \mathbb{N}$ telle que $\phi(x) = 0$ si et seulement si $x = 1_M$, et pour tous $x, y \in M$,

$$\phi(x \cdot y) = \phi(x) + \phi(y). \quad (1.1.7)$$

En d'autres termes, M est gradué s'il admet un morphisme vers le monoïde $(\mathbb{N}, +, 0)$ de telle sorte que le seul antécédent de 0 est 1_M . Lorsqu'il existe une application $\cdot^{-1} : M \rightarrow M$ telle que, pour tout $x \in M$,

$$x \cdot_M x^{-1} = 1_M = x^{-1} \cdot_M x, \quad (1.1.8)$$

le monoïde M est un *groupe*. Dans ce cas, si x est un élément de M , l'élément x^{-1} est dit *l'inverse* de x .

Voici quelques exemples de monoïdes :

1. le *monoïde des entiers naturels* $(\mathbb{N}, +, 0)$ est gradué par l'application identité, et finiment engendré par l'ensemble générateur $\{1\}$.
2. le *monoïde additif* $(\mathbb{Z}, +, 0)$ est un groupe, l'inverse d'un élément étant son opposé. Le monoïde des entiers naturels est un sous-monoïde du monoïde additif.
3. Le *monoïde multiplicatif* $(\mathbb{N}, \times, 1)$ est gradué par l'application qui associe à chaque élément le nombre de facteurs premiers qui le composent. De plus, ce monoïde n'est pas finiment engendré.
4. Le monoïde \max $(\mathbb{N}, \max, 0)$ n'est ni gradué, ni finiment engendré.
5. Pour tout $n \geq 1$, le monoïde $(\{E \subseteq [n]\}, \cap, [n])$, où $[n]$ désigne l'ensemble $\{1, \dots, n\}$ est fini et n'est pas gradué.
6. Pour tout $n \geq 1$, le monoïde $(\{E \subseteq [n]\}, \cup, \emptyset)$ est également fini et non gradué.

1.1.3 Ensembles partiellement ordonnés et treillis

Dans cette partie, nous rappelons les notions de base concernant les ensembles partiellement ordonnés et les treillis que nous utiliserons dans ce mémoire. Pour des notions plus précises et détaillées, le lecteur pourra consulter [BS81] ou [Sta11].

Définition 1.1.3. Un ensemble partiellement ordonné — ou poset — est un ensemble P muni d'une relation partielle \leq vérifiant les conditions

- (i) de réflexivité : $x \leq x$ pour tout $x \in P$;
- (ii) de transitivité : $x \leq y$ et $y \leq z$ impliquent $x \leq z$ pour tous $x, y, z \in P$;
- (iii) d'antisymétrie : $x \leq y$ et $y \leq x$ impliquent $x = y$ pour tous $x, y \in P$.

Posons quelques notions de vocabulaire à propos des posets. Soit (P, \leq_P) un poset. Le poset P est *fini* si l'ensemble P est fini. Un ensemble P' est un *sous-poset* de P si $P' \subseteq P$ et P' est muni de la relation d'ordre de P restreinte aux éléments de P' . La relation d'ordre \leq_P est *totale* si pour tous $x, y \in P$, nous avons $x \leq_P y$ ou $y \leq_P x$. Soit (Q, \leq_Q) un autre poset. Si P et Q sont égaux en tant qu'ensembles et que pour tous $x, y \in P$,

$$x \leq_P y \quad \text{implique} \quad x \leq_Q y, \quad (1.1.9)$$

alors Q est une *extension* de P . Si la relation d'ordre \leq_Q est de plus totale, alors Q est une *extension linéaire* de P . L'*intervalle* $[x, z]$, où $x, z \in P$, est l'ensemble

$$[x, z] := \{y \in P : x \leq_P y \leq_P z\}. \quad (1.1.10)$$

L'ensemble des *couvertures* de $x \in P$ est l'ensemble

$$\{y \in P : \#[x, y] = 2\}. \quad (1.1.11)$$

Un sous-ensemble E de P est *clos par intervalle* si pour tous $x, y \in E$, $[x, y] \subseteq E$. Une suite d'éléments x_1, x_2, \dots, x_ℓ de P est une *chaîne* si

$$x_1 \leq_P x_2 \leq_P \dots \leq_P x_\ell ; \quad (1.1.12)$$

cette chaîne est de plus *saturée* si x_{i+1} couvre x_i pour tout $i \in [\ell - 1]$. Si l'élément $x \in P$ vérifie pour tout $y \in P$,

$$x \leq_P y \quad \text{implique} \quad x = y, \quad (1.1.13)$$

alors x est un *élément maximal* de P . Si l'élément $y \in P$ vérifie pour tout $x \in P$, $x \leq_P y$, alors y est le *plus grand élément* de P . On définit de la même manière la notion d'*élément minimal* et de *plus petit élément* en inversant la relation d'ordre \leq_P . Si P' est un sous-poset P , la *borne supérieure* de P' , notée $\sup P'$, est le plus petit élément, s'il existe, du sous-poset de P constitué des éléments

$$\{y \in P : \text{pour tout } x \in P', x \leq_P y\}. \quad (1.1.14)$$

De même, la *borne inférieure* de P' , notée $\inf P'$, est le plus grand élément, s'il existe, du sous-poset de P

$$\{x \in P : \text{pour tout } y \in P', x \leq_P y\}. \quad (1.1.15)$$

S'il existe une application $\phi : P \rightarrow \mathbb{N}$ telle que $\phi(x) = 0$ si x est un élément minimal de P , et $\phi(y) = \phi(x) + 1$ si y couvre x dans P , alors P est *gradué*. Le *diagramme de Hasse* de P est le graphe orienté (voir le paragraphe 1.3.1) dont l'ensemble de sommets est P et dont les arcs correspondent aux couples (x, y) tel que y couvre x dans P . Pour alléger nos représentations graphiques, les arcs d'un diagramme de Hasse ne seront pas orientés ; ils le sont implicitement par le fait que l'élément couvert se trouve plus haut que l'élément couvrant.

Certains posets admettent une structure plus fine :

Définition 1.1.4. Un treillis (L, \leq) est un poset tel que pour tous $x, y \in L$, l'ensemble $\{x, y\}$ admet une borne inférieure et une borne supérieure.

Une définition des treillis, alternative à la définition 1.1.4, propose de voir une telle structure comme un ensemble L muni de deux lois de composition internes \vee et \wedge qui vérifient les conditions

- (i) d'associativité, de commutativité et d'idempotence (*i.e.*, $x \vee x = x$ et $x \wedge x = x$ pour tout $x \in L$);
- (ii) d'absorption : $x \vee (x \wedge y) = x \wedge (x \vee y)$ pour tous $x, y \in L$.

À partir de la donnée d'un triplet (L, \vee, \wedge) , nous pouvons construire sur L une relation d'ordre \leq , définie pour tous $x, y \in L$ par

$$x \leq y \quad \text{si et seulement si} \quad x \vee y = y \quad \text{et} \quad x \wedge y = x. \quad (1.1.16)$$

Inversement, à partir d'un treillis (L, \leq) , nous obtenons la structure précédemment définie en posant pour tous $x, y \in L$,

$$x \vee y := \sup\{x, y\} \quad \text{et} \quad x \wedge y := \inf\{x, y\}. \quad (1.1.17)$$

Voici quelques exemples de posets et de treillis :

1. l'ensemble des entiers naturels \mathbb{N} , muni de la relation d'ordre usuelle \leq , est un poset. Il possède un plus petit élément, 0, mais pas de plus grand élément. Ce poset est de plus totalement ordonné et il s'agit ainsi d'un treillis, où les opérateurs \vee et \wedge correspondants vérifient $x \vee y := \max\{x, y\}$ et $x \wedge y := \min\{x, y\}$. Son diagramme de Hasse est de la forme

$$\begin{array}{c} 0 \\ | \\ 1 \\ | \\ 2 \\ \vdots \end{array} \quad (1.1.18)$$

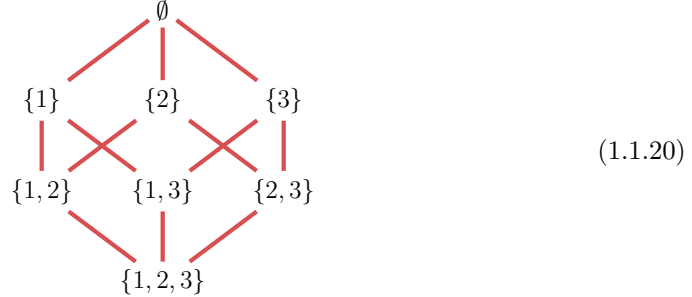
2. L'ensemble $[n]$, muni de la relation d'ordre $|$ où l'on a $x|y$ si x divise y , est pour tout $n \geq 1$ un poset fini. Son plus petit élément est 1 et si $n \geq 3$, il ne possède pas de plus grand élément. En effet, les éléments n et $n-1$ sont maximaux et non comparables pour $|$. Le diagramme de Hasse pour le cas $n := 8$ est

$$\begin{array}{ccccc} & & 1 & & \\ & \swarrow & & \searrow & \\ 2 & & 3 & & 5 & & 7 \\ | & \swarrow & \searrow & & & & \\ 4 & & 6 & & & & \\ | & & & & & & \\ 8 & & & & & & \end{array} \quad (1.1.19)$$

Lorsque l'on munit \mathbb{N} de la relation $|$ on obtient un treillis où les opérateurs \vee et \wedge correspondants vérifient $x \vee y := \text{ppcm}(x, y)$ et $x \wedge y := \text{pgcd}(x, y)$.

3. L'ensemble $\{E \subseteq [n]\}$ admet pour tout $n \geq 1$ une structure de treillis fini pour les opérateurs \vee et \wedge définis par $E \vee F := E \cup F$ et $E \wedge F := E \cap F$. Ce treillis est connu sous

le nom de *treillis booléen* ou *hypercube*. La relation d'ordre correspondant à \vee et \wedge est la relation \subseteq d'inclusion ensembliste. Le diagramme de Hasse pour le cas $n := 3$ est



1.1.4 Espaces vectoriels combinatoires

Pour le reste de ce mémoire, le symbole \mathbb{K} désigne un corps quelconque de caractéristique nulle. Sauf mention contraire, toutes les structures algébriques le nécessitant sont définies sur \mathbb{K} , et ceci implicitement.

Nous supposons connues du lecteur les notions basiques d'algèbre linéaire. Le but de ce paragraphe est d'introduire le vocabulaire que nous utiliserons dans le reste du texte, notamment dans le contexte des espaces vectoriels libres sur un ensemble. En effet, la plupart des espaces vectoriel que nous manipulerons sont des espaces vectoriels dont les bases sont indexées par des objets combinatoires.

Le *symbole de Kronecker*, noté $\delta_{x,y}$, est dans notre contexte à valeurs dans \mathbb{K} ; et $\delta_{x,y} := 1_{\mathbb{K}}$ si $x = y$, et $\delta_{x,y} := 0_{\mathbb{K}}$ sinon, où $1_{\mathbb{K}}$ désigne l'élément neutre pour le produit de \mathbb{K} et $0_{\mathbb{K}}$ — ou simplement 0 — désigne l'élément neutre pour l'addition.

Rappelons une construction qui à tout ensemble E associe un espace vectoriel. On appelle *support* d'une application $f : E \rightarrow \mathbb{K}$ l'ensemble

$$\text{Supp}(f) := \{x \in E : f(x) \neq 0\}. \quad (1.1.21)$$

Soit

$$\text{Vect}(E) := \{f : E \rightarrow \mathbb{K} : f \text{ est à support fini}\}. \quad (1.1.22)$$

L'ensemble $\text{Vect}(E)$ admet naturellement une structure d'espace vectoriel : si $f, g \in \text{Vect}(E)$, alors $f + g$ est l'application $x \mapsto f(x) + g(x)$, et de même, si $\lambda \in \mathbb{K}$, alors λf est l'application $x \mapsto \lambda f(x)$. Nous appelons $\text{Vect}(E)$ *l'espace vectoriel libre* sur E . L'ensemble

$$\{\mathbf{F}_x := y \mapsto \delta_{x,y} : x \in E\} \quad (1.1.23)$$

forme une base évidente de $\text{Vect}(E)$. Ainsi, tout élément f de $\text{Vect}(E)$ s'exprime sous la forme

$$f = \sum_{x \in \text{Supp}(f)} f(x) \mathbf{F}_x, \quad (1.1.24)$$

et nous pouvons voir $\text{Vect}(E)$ comme l'espace vectoriel des sommes formelles finies d'éléments de E à coefficients dans \mathbb{K} . La base des \mathbf{F} est la *base fondamentale* de $\text{Vect}(E)$. Remarquons que dans certains cas, par souci de lisibilité, nous noterons simplement par x un élément \mathbf{F}_x de $\text{Vect}(E)$.

Nous rencontrerons dans ce mémoire des espaces vectoriels sous une forme particulière :

Définition 1.1.5. Soit C une classe combinatoire telle que $\#C_0 = 1$. L'espace vectoriel combinatoire engendré par C est l'espace vectoriel libre $\text{Vect}(C)$. Nous dirons que C est la classe combinatoire sous-jacente à $\text{Vect}(C)$.

Tout espace vectoriel combinatoire est *gradué*, c'est-à-dire qu'il se décompose comme une somme directe de la manière suivante :

$$\text{Vect}(C) = \bigoplus_{n \geq 0} \text{Vect}(C_n). \quad (1.1.25)$$

D'après la définition 1.1.1 des classes combinatoires, les espaces vectoriels $\text{Vect}(C_n)$, appelés *composantes homogènes de degré n* de $\text{Vect}(C)$, sont de dimension finie. Si V est un espace vectoriel combinatoire, nous emploierons la notation $V^{(n)}$ pour faire référence à sa composante homogène de degré n .

La *série de Hilbert* d'un espace vectoriel combinatoire V est la série

$$F_V(t) := \sum_{n \geq 0} \dim V^{(n)} t^n, \quad (1.1.26)$$

qui n'est rien d'autre que la série génératrice de la classe combinatoire sous-jacente à V .

Soient maintenant V et W deux espaces vectoriels combinatoires. On rappelle qu'une *application linéaire* $\phi : V \rightarrow W$ est une application qui vérifie pour tous $\lambda \in \mathbb{K}$, et $x, y \in V$,

$$\phi(\lambda x) = \lambda \phi(x) \quad \text{et} \quad \phi(x + y) = \phi(x) + \phi(y). \quad (1.1.27)$$

Une *forme linéaire* sur V est une application linéaire $\phi : V \rightarrow \mathbb{K}$ où \mathbb{K} est vu comme un espace vectoriel.

Soient V et W deux espaces vectoriels combinatoires engendrés respectivement par les classes combinatoires C et D . Le *produit tensoriel* de V avec W , noté $V \otimes W$, est l'espace vectoriel combinatoire tel que si $\{E_c : c \in C\}$ est une base de V et $\{F_d : d \in D\}$ est une base de W , alors

$$\{E_c \otimes F_d : (c, d) \in C \times D\} \quad (1.1.28)$$

est une base de $V \otimes W$. Ainsi, les éléments de cet espace vectoriel s'écrivent sur cette base comme combinaisons linéaires d'éléments de la forme $E_c \otimes F_d$. On appelle *tenseurs* de tels éléments. L'espace vectoriel $V \otimes W$ est en outre muni, pour tous $\lambda \in \mathbb{K}$, $x, y \in V$, et $z, t \in W$, des relations suivantes :

$$x \otimes (z + t) = x \otimes z + x \otimes t, \quad (1.1.29)$$

$$(x + y) \otimes z = x \otimes z + y \otimes z, \quad (1.1.30)$$

$$(\lambda x) \otimes z = x \otimes (\lambda z). \quad (1.1.31)$$

Soient V, V', W, W' des espaces vectoriels combinatoires et $\phi : V \rightarrow W$ et $\phi' : V' \rightarrow W'$ des applications linéaires. Le *produit tensoriel* de ϕ et ϕ' , noté $\phi \otimes \phi'$, est l'application linéaire

$$\phi \otimes \phi' : V \otimes V' \rightarrow W \otimes W' \quad (1.1.32)$$

définie pour tous $x \in V$, $y \in V'$, par

$$\phi \otimes \phi'(x \otimes y) := \phi(x) \otimes \phi'(y). \quad (1.1.33)$$

Rappelons que l'*espace vectoriel dual* d'un espace vectoriel V de dimension finie est constitué de l'ensemble de ses formes linéaires. Le *dual gradué* — ou simplement *dual* — d'un espace vectoriel combinatoire V est la somme directe des espaces vectoriels duaux de chacune des composantes homogènes de V . Si $\{E_c : c \in C\}$ est une base de V , nous noterons par $\{F_c^* : c \in C\}$ sa *base duale*. Le *crochet de dualité* est la forme linéaire $\langle \cdot, \cdot \rangle : V \otimes V^* \rightarrow \mathbb{K}$ définie par $\langle E_c, F_d^* \rangle := \delta_{c,d}$.

1.2 Combinatoire des mots et des permutations

Pour le reste de ce texte, $A := \{a_1, a_2, a_3, \dots\}$ est un alphabet infini et totalement ordonné par $a_i \leq a_j$ si et seulement si $i \leq j$.

1.2.1 Mots et opérations sur les mots

Un *mot* sur l'alphabet A est une suite finie de lettres de A . Sauf mention contraire, tous les mots que nous considérerons seront sur cet alphabet. La *taille* $|u|$ d'un mot u , encore appelée sa *longueur*, est le nombre de lettres qui le constituent. L'unique mot de longueur 0 est le *mot vide* et est noté ϵ . La notation u_i désigne la i^{e} lettre de u , pour tout $i \in [|u|]$. Soit $u := u_1 u_2 \dots u_{|u|}$ un mot. Un *facteur* de u est un mot de la forme $u_i u_{i+1} \dots u_j$ avec $i \leq j \in [|u|]$. Lorsque $i = 1$, ce facteur est également un *préfixe* de u , et lorsque $j = |u|$, il s'agit d'un *suffixe*. Un *sous-mot* de u est un mot de la forme $u_{i_1} u_{i_2} \dots u_{i_k}$ où

$$1 \leq i_1 < i_2 < \dots < i_k \leq |u|. \quad (1.2.1)$$

On note par $\text{Alph}(u)$ l'ensemble $\{u_i : i \in [|u|]\}$, qui est le plus petit alphabet sur lequel u est défini. Si E est un sous-ensemble de A , on note par $u|_E$ la *restriction de u à l'alphabet E* , c'est-à-dire le plus long sous-mot u' de u tel que $\text{Alph}(u') = E$. L'*évaluation* de u est le mot

$$\text{ev}(u) := a_1^{|u|_{a_1}} a_2^{|u|_{a_2}} \dots a_\ell^{|u|_{a_\ell}}, \quad (1.2.2)$$

où $|u|_{a_i} := |u|_{\{a_i\}}$ dénote le *nombre d'occurrences* de la lettre a_i dans u . La plus grande (resp. plus petite) lettre de u est notée $\max(u)$ (resp. $\min(u)$). Si u admet un sous-mot de la forme $u_i u_j$ avec $u_i > u_j$, le couple (i, j) est une *inversion* de u . De plus, si $j = i + 1$, l'indice i est qualifié de *descente*.

L'ensemble des mots sur A peut être muni de deux ordre totaux. Le premier, appelé *ordre lexicographique*, est noté \leq_{lex} et vérifie pour tous mots u et v sur A ,

$$u \leq_{\text{lex}} v \quad \text{si et seulement si} \quad \begin{array}{l} u \text{ est un préfixe de } v \\ \text{ou} \\ u = w a u' \text{ et } v = w b v' \text{ avec } a < b. \end{array} \quad (1.2.3)$$

Le second, appelé *ordre lexicographique par longueur*, est noté \leq_{glex} et vérifie pour tous mots u et v sur A ,

$$u \leq_{\text{glex}} v \quad \text{si et seulement si} \quad \begin{array}{l} |u| < |v| \\ \text{ou} \\ |u| = |v| \text{ et } u \leq_{\text{lex}} v. \end{array} \quad (1.2.4)$$

Une opération naturelle sur les mots, à savoir la *concaténation*, consiste à les placer bout à bout. Si $u := u_1 \dots u_n$ et $v := v_1 \dots v_m$ sont deux mots, la concaténation de u et v , notée $u \cdot v$, est le mot $u_1 \dots u_n v_1 \dots v_m$. Cette opération munit l'ensemble des mots d'une structure de monoïde, le *monoïde libre*, que l'on note A^* . Ce monoïde admet ϵ comme élément neutre et est gradué par l'application qui à un mot associe sa longueur. Il est de plus engendré par l'ensemble A . Nous notons par A^+ l'ensemble des mots non vides et par A^n l'ensemble des mots de longueur n sur A . De plus, $\mathbb{K}\langle A \rangle$ désigne l'espace vectoriel engendré par les mots de A^* , ou, de manière équivalente, l'*espace vectoriel des polynômes non commutatifs* dont les variables sont des éléments de A . De manière générale, les notations E^* , E^+ , E^n et $\mathbb{K}\langle E \rangle$ sont valables pour un ensemble E quelconque, et désignent respectivement le monoïde libre des suites finies d'éléments de E , l'ensemble des suites finies et non vides d'éléments de E , l'ensemble des suites de longueur n d'éléments de E et l'espace vectoriel des polynômes non commutatifs sur l'ensemble E de variables.

Rappelons maintenant quelques opérations sur les mots. L'*image miroir* de u est le mot

$$u^\sim := u_{|u|} \dots u_2 u_1. \quad (1.2.5)$$

Soit v un mot. Le *produit de mélange* \sqcup de u et de v est défini linéairement sur $\mathbb{K}\langle A \rangle$ par

$$u \sqcup v := \begin{cases} u & \text{si } v = \epsilon, \\ v & \text{si } u = \epsilon, \\ \mathbf{a}(u' \sqcup \mathbf{b}v') + \mathbf{b}(\mathbf{a}u' \sqcup v') & \text{sinon, où } u = \mathbf{a}u' \text{ et } v = \mathbf{b}v'. \end{cases} \quad (1.2.6)$$

Par exemple,

$$\begin{aligned} \mathbf{a}_1 \mathbf{a}_2 \sqcup \mathbf{a}_2 \mathbf{a}_1 &= \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_2 \mathbf{a}_1 + \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_1 + \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_2, \\ &= \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 + 2\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_2 \mathbf{a}_1 + 2\mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_1. \end{aligned} \quad (1.2.7)$$

Soit $\cdot^\#$ la transformation qui à un alphabet totalement ordonné $B := \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots\}$ associe l'alphabet $B^\# := \{\mathbf{b}_1^\#, \mathbf{b}_2^\#, \mathbf{b}_3^\#, \dots\}$ ordonné par $\mathbf{b}_i^\# \leq \mathbf{b}_j^\#$ si et seulement si $\mathbf{b}_j \leq \mathbf{b}_i$. En imposant l'égalité $\mathbf{b}_i^{\#\#} = \mathbf{b}_i$, la transformation $\cdot^\#$ est une involution sur les alphabets, *i.e.*, $B^{\#\#} = B$. L'*involution de Schützenberger*, également notée $\cdot^\#$ est une involution sur A^* définie par

$$u^\# := u_{|u|}^\# \dots u_2^\# u_1^\#. \quad (1.2.8)$$

Par exemple, $(\mathbf{a}_5 \mathbf{a}_3 \mathbf{a}_1 \mathbf{a}_5 \mathbf{a}_2)^\# = \mathbf{a}_2^\# \mathbf{a}_5^\# \mathbf{a}_1^\# \mathbf{a}_3^\# \mathbf{a}_5^\#$.

Dans un souci d'alléger les notations, les mots seront notés par des suites d'entiers, spécifiant sans ambiguïté les lettres de A correspondantes. Ainsi, à titre d'exemple, la suite 213211 sera la notation du mot $\mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_1$.

1.2.2 Permutations, opérations sur les permutations et permutoèdre

Une *permutation* est une bijection $\sigma : [n] \rightarrow [n]$. Nous adopterons la *notation mot* des permutations; σ sera en effet notée par le mot $\sigma(1)\sigma(2)\dots\sigma(n)$. Les permutations sont donc vues comme des mots u qui vérifient $\text{Alph}(u) = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{|u|}\}$, et ainsi, les définitions du paragraphe précédent s'y appliquent. La *taille* d'une permutation est la longueur du mot qui la représente. La classe combinatoire des permutations est notée \mathfrak{S} . Si σ est de la forme $\sigma = u\mathbf{a}_{i+1}v\mathbf{a}_i w$, où $\mathbf{a}_i, \mathbf{a}_{i+1} \in A$ et u, v et w sont des mots, alors \mathbf{a}_i est un *recul* de σ .

Chaque ensemble \mathfrak{S}_n est muni d'une relation d'ordre partielle \leq_p engendrée par la relation de couverture qui change un facteur de la forme \mathbf{ab} d'une permutation en le facteur \mathbf{ba} sous la condition que $\mathbf{a} < \mathbf{b}$. Autrement dit, une permutation σ est couverte par une permutation ν si σ est de la forme $\sigma = u\mathbf{ab}v$ et ν de la forme $\nu = u\mathbf{bav}$. Cet poset est en fait un treillis, et est connu sous le nom de *permutoèdre droit* [GR63]. Nous l'appellerons dans ce qui suit simplement *permutoèdre*. La figure 1.1 montre le diagramme de Hasse du permutoèdre droit d'ordre 4.

Chaque ensemble \mathfrak{S}_n possède une structure de groupe où le produit de deux permutations est leur composition lorsque celles-ci sont vues comme des applications. L'inverse d'une permutation est son application inverse. Ce groupe est connu sous le nom de *groupe symétrique*. Nous avons ainsi par exemple $35412 \cdot 41253 = 13524$ et $35412^{-1} = 45132$. Une permutation σ qui vérifie $\sigma^{-1} = \sigma$ est une *involution*. Le groupe symétrique \mathfrak{S}_n admet la présentation suivante : c'est le groupe engendré par l'ensemble $\{s_1, \dots, s_{n-1}\}$ soumis aux relations

$$s_i^2 = 1, \quad (1.2.9)$$

$$s_i \cdot s_j = s_j \cdot s_i \quad \text{pour tous } |i - j| \geq 2, \quad (1.2.10)$$

$$s_i \cdot s_{i+1} \cdot s_i = s_{i+1} \cdot s_i \cdot s_{i+1} \quad \text{pour tout } i \in [n - 2], \quad (1.2.11)$$

où 1 désigne l'élément neutre du groupe. Les éléments s_i sont appelés *transposition élémentaires* et agissent à droite sur les permutations par transposition de lettres. Nous avons ainsi

$$\sigma \cdot s_i = \sigma_1 \dots \sigma_{i+1} \sigma_i \dots \sigma_n. \quad (1.2.12)$$

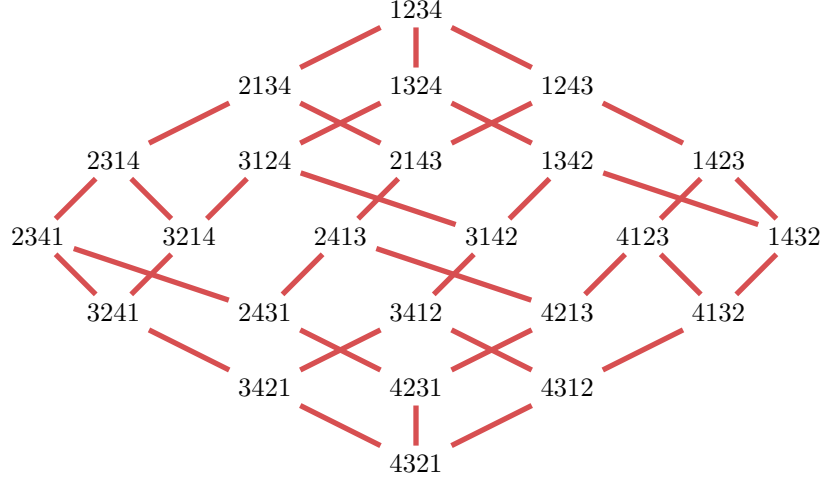


FIGURE 1.1 – Le diagramme de Hasse du permutoèdre droit d'ordre 4.

pour toute permutation $\sigma \in \mathfrak{S}_n$ et $i \in [n-1]$.

Rappelons maintenant quelques opérations classiques sur les permutations. Soient $\sigma \in \mathfrak{S}_n$ et $\nu \in \mathfrak{S}_m$. L'opération *over* \diagup est définie par

$$\sigma \diagup \nu := \sigma_1 \dots \sigma_n(\nu_1+n) \dots (\nu_m+n). \quad (1.2.13)$$

L'opération *under* \diagdown est quant à elle définie par

$$\sigma \diagdown \nu := (\nu_1+n) \dots (\nu_m+n) \sigma_1 \dots \sigma_n. \quad (1.2.14)$$

Par exemple, $\textcolor{red}{312} \diagup 2314 = \textcolor{red}{3125647}$ et $\textcolor{red}{312} \diagdown 2314 = 5647\textcolor{red}{312}$. La permutation σ est *connexe* si $\sigma = \sigma^{(1)} \diagup \sigma^{(2)}$ implique $\sigma^{(1)} = \sigma$ ou $\sigma^{(2)} = \sigma$. De même, la permutation σ est *anti-connexe* si $\sigma = \sigma^{(1)} \diagdown \sigma^{(2)}$ implique $\sigma^{(1)} = \sigma$ ou $\sigma^{(2)} = \sigma$. Par exemple, la permutation 31254 n'est pas connexe puisque $31254 = 312 \diagup 21$, et la permutation 35421 n'est pas anti-connexe puisque $35421 = 21 \diagdown 132$. En revanche, 35421 est connexe et 31254 est anti-connexe. Le *produit de mélange décalé* \sqcup est défini linéairement sur l'espace vectoriel combinatoire $\text{Vect}(\mathfrak{S})$ par

$$\sigma \sqcup \nu := \sigma \sqcup ((\nu_1+|\sigma|) \dots (\nu_{|\nu|}+|\sigma|)). \quad (1.2.15)$$

Par exemple,

$$\textcolor{red}{12} \sqcup 21 = \textcolor{red}{12} \sqcup 43 = \textcolor{red}{1243} + \textcolor{red}{1423} + \textcolor{red}{1432} + \textcolor{red}{4123} + \textcolor{red}{4132} + 4312. \quad (1.2.16)$$

La *standardisation* std est une application surjective de l'ensemble des mots vers l'ensemble des permutations qui associe à un mot u l'unique permutation $\text{std}(u) \in \mathfrak{S}_{|u|}$ qui possède les mêmes inversions que u . Par exemple, $\text{std}(314257423) = 416289735$.





1.3 Combinatoire des graphes et des arbres

Nous avons besoin de définitions très basiques à propos des graphes. Après avoir rappelé celles-ci, nous introduisons le vocabulaire et autres définitions de base concernant les structures arborescentes qui vont intervenir dans ce mémoire.

1.3.1 Graphes


Un *graphe* est un couple (V, E) où V est un ensemble et E un ensemble de paires d'éléments de V . Les éléments de V sont appelés *sommets* et ceux de E , *arêtes*. Soit $G := (V, E)$ un graphe. Un *chemin* est une suite de sommets de G de la forme (x_1, \dots, x_ℓ) telle que $\{x_i, x_{i+1}\} \in E$ pour tout $i \in [\ell - 1]$. Nous dirons que ce chemin *connecte* le sommet x_1 au sommet x_ℓ et que sa *longueur* est ℓ . Ce chemin est aussi un *cycle* si $\{x_\ell, x_1\} \in E$. Le graphe G est *connexe* si pour tous sommets $x, x' \in V$, il existe au moins un chemin connectant x à x' . Tout sous-ensemble V' de V induit un *sous-graphe* de G : c'est le graphe dont les sommets sont ceux de V' , et les arêtes, celles de la forme $\{x, y\}$ où $\{x, y\} \in E$ et $x, y \in V'$. Une *composante connexe* de G est un sous-graphe connexe maximal de G . Le graphe G est *acyclique* s'il ne possède aucun cycle. Un *graphe étiqueté* est un graphe muni d'une application $\tau : V \rightarrow A$ qui associe à chaque sommet x un élément de l'alphabet A .

Un graphe est *orienté* s'il est muni de deux applications $\alpha : E \rightarrow V$ et $\beta : E \rightarrow V$ telles que $\alpha(\{x, y\}) = x$ si et seulement si $\beta(\{x, y\}) = y$. L'application α associe à une arête son *origine*, tandis que l'application β lui associe son *but*. Une arête dans un graphe orienté est appelée *arc*. Un *chemin orienté* dans un graphe orienté est un chemin (x_1, \dots, x_ℓ) tel que $\beta(\{x_i, x_{i+1}\}) = \alpha(\{x_{i+1}, x_{i+2}\})$ pour tout $i \in [\ell - 2]$. Un sommet x d'un graphe orienté est une *source* si pour tout autre sommet y , il existe un chemin qui connecte x à y . Dans certains cas, il est plus approprié de voir un graphe orienté comme un couple (V, E) où E est un ensemble fini de sommets et V un ensemble de couples de sommets. Le couple $(x, y) \in E$ indique en effet que le graphe contient une arête $\{x, y\}$ telle que $\alpha(\{x, y\}) = x$ et $\beta(\{x, y\}) = y$.

Dans nos représentations graphiques des graphes, les sommets seront dessinés par des cercles  et les arêtes par des segments  reliant deux sommets. Les arêtes d'un graphe orienté sont munies d'une flèche  qui pointe vers leur but. Un chemin arbitraire sera représentée par une ligne en zigzag  reliant les deux sommets qu'il connecte. L'étiquette d'un sommet d'un graphe étiqueté sera dessinée à l'intérieur du cercle qui le représente.

1.3.2 Structures arborescentes

Un *arbre* est un graphe à la fois connexe et acyclique. Il est d'usage courant d'appeler *nœuds* les sommets d'un arbre et *branches* ses chemins. Un *arbre enraciné* est un couple (G, r) où G est un arbre et r est un nœud de G appelé *racine*. Soit T un arbre enraciné de racine r , et x et y des nœuds de T . Si le chemin qui connecte r à y contient x , alors x est un *ancêtre* de y . Dans ce cas y est également un *descendant* de x . En particulier, la racine r est ancêtre de tous les nœuds de T . Le nœud x est le *parent* de y si x est un ancêtre de y et $\{x, y\}$ est une arête de T . Dans ce cas, y est également un *fil* de x . Le *sous-arbre* T' de racine x dans T est l'arbre enraciné $((V', E'), x)$ où V' est le sous-ensemble des nœuds de V qui sont descendants de x , et E' le sous-ensemble des arêtes $\{y, z\}$ de E telles que $y, z \in V'$. Un *arbre plan* est un arbre enraciné muni d'une application $\omega : V \rightarrow V^*$ telle que pour tous nœuds x et y , le mot $\omega(x)$ est sans répétition de lettres, et $\omega(x)$ admet une occurrence de y si et seulement si y est un fil de x . Ainsi, ω impose un ordre total sur les fils des nœuds de l'arbre. Si x possède ℓ fils, x est *d'arité* ℓ , et le nœud $\omega(x)_1$ est *le fils le plus à gauche* de x , et le nœud $\omega(x)_\ell$ est *le fils le plus à droite* de x . Si T_1, T_2, \dots, T_ℓ sont des arbres enracinés, la *greffe* de ces arbres, notée $\wedge(T_1, T_2, \dots, T_\ell)$, est l'arbre enraciné qui possède comme sous-arbres T_1, T_2, \dots, T_ℓ , de gauche à droite.

Nous aurons besoin dans certains cas dans nos représentations graphiques de dessiner des sous-arbres arbitraires. Nous les schématiserons par des carrés . Dans le dessin d'un arbre enraciné, la racine est toujours le nœud dessiné le plus en haut. Les arbres plans seront dessinés de telle sorte que, de gauche à droite, les fils d'un nœuds x sont représentés conformément à l'ordre total imposé par le mot $\omega(x)$.

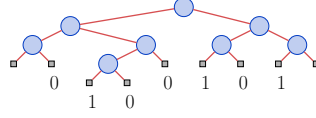




FIGURE 1.3 – Un arbre binaire dont la canopée est 0100101.

dû à Viennot [Vie04] mais la notion, elle, fut déjà utilisée en [LR98]. La figure 1.3 montre un exemple de canopée.

Rappelons quelques opérations sur les arbres binaires. Soient T_0 et T_1 deux arbres binaires. L'opérateur de *greffe* d'arbres binaires est noté dans ce contexte de manière infixée, et ainsi, $T_0 \wedge T_1$ est l'arbre binaire ayant T_0 comme sous-arbre gauche et T_1 comme sous-arbre droit. Si T_0 et T_1 sont étiquetés, on note $T_0 \wedge_a T_1$ l'arbre binaire ayant T_0 comme sous-arbre gauche et T_1 comme sous-arbre droit, et où la racine est étiquetée par la lettre a . Les opérateurs *over* $/$ et *under* \backslash de Loday et Ronco [LR02] se définissent comme suit. L'arbre binaire T_0 / T_1 est obtenu en remplaçant la feuille la plus à gauche de T_1 par la racine de T_0 . Similairement, $T_0 \backslash T_1$ est obtenu en remplaçant la feuille la plus à droite de T_0 par la racine de T_1 . Voici quelques exemples pour $T_0 :=$  et $T_1 :=$ .

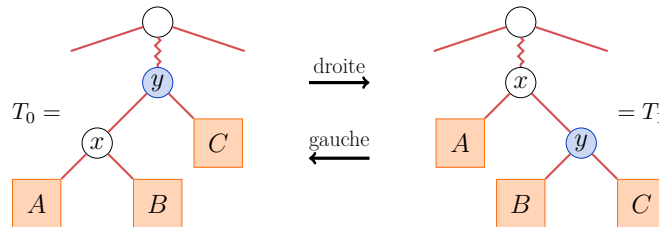
$$T_0 \wedge T_1 = \text{arbre binaire avec } T_0 \text{ à gauche et } T_1 \text{ à droite}, \quad (1.3.4)$$

$$T_0 / T_1 = \text{arbre binaire obtenu en remplaçant la feuille la plus à gauche de } T_1 \text{ par la racine de } T_0, \quad (1.3.5)$$

$$T_0 \backslash T_1 = \text{arbre binaire obtenu en remplaçant la feuille la plus à droite de } T_0 \text{ par la racine de } T_1. \quad (1.3.6)$$

Rappelons un autre opération fondamentale :

Définition 1.3.1. Soit T_0 un arbre binaire et y un nœud de T_0 ayant un sous-arbre gauche non vide. Soit $S_0 := (A \wedge B) \wedge C$ le sous-arbre de racine y dans T_0 et T_1 l'arbre binaire obtenu en remplaçant S_0 par $A \wedge (B \wedge C)$ (voir la figure 1.4). Alors, la rotation droite en y envoie T_0 sur T_1 .

FIGURE 1.4 – L'opération de rotation dans un arbre binaire. La rotation droite en y envoie l'arbre binaire T_0 sur l'arbre binaire T_1 . La rotation gauche en y envoie quant à elle T_1 sur T_0 .

Si une rotation droite envoie un arbre binaire T_0 sur un arbre binaire T_1 , alors T_0 est en *relation partielle de Tamari* avec T_1 . Cette relation est notée $T_0 \prec T_1$. De plus, on notera $T_0 \prec_y T_1$ si c'est une rotation droite en le nœud y de T_0 qui envoie T_0 sur T_1 . Tant qu'il n'y a pas de confusion, le terme *rotation* signifiera *rotation droite* dans la suite.

On peut munir chaque ensemble \mathcal{T}_n d'une structure de treillis : le treillis de Tamari [Tam62]. Ce treillis peut être défini par l'intermédiaire d'objets combinatoires en bijection avec les arbres

binaires. La plupart des définitions portent sur des vecteurs d'entiers vérifiant certaines conditions [Sta99a], sur des forêts et des arbres binaires [Knu04] et sur des chemins de Dyck [BB09].

Définition 1.3.2. La relation d'ordre de Tamari $\leq_{\mathcal{T}}$ est la clôture réflexive et transitive de la relation partielle de Tamari \prec . En d'autres termes, si T_0 et T_ℓ sont deux arbres binaires de même taille, nous avons $T_0 \leq_{\mathcal{T}} T_\ell$ si $\ell \geq 0$ et il existe une suite $T_1, \dots, T_{\ell-1}$ d'arbres binaires telle que

$$T_0 \prec T_1 \prec \dots \prec T_{\ell-1} \prec T_\ell. \quad (1.3.7)$$

Pour tout $n \geq 0$, on note \mathbb{T}_n le poset $(\mathcal{T}_n, \leq_{\mathcal{T}})$.

La relation de Tamari est bien une relation d'ordre. En effet, $\leq_{\mathcal{T}}$ est réflexive et transitive par définition. Pour montrer que $\leq_{\mathcal{T}}$ est antisymétrique, considérons la statistique $\phi : \mathcal{T} \rightarrow \mathbb{N}$ où $\phi(T)$ est la somme, pour chaque nœud x de T , du nombre de nœuds dans le sous-arbre droit de x . Il est immédiat que si $T_0 \prec T_1$ alors $\phi(T_0) < \phi(T_1)$, ce qui montre que $\leq_{\mathcal{T}}$ est antisymétrique.

Huang et Tamari [HT72] ont montré que les posets \mathbb{T}_n forment un treillis. Ce sont les *treillis de Tamari*. La figure 1.5 montre les diagrammes de Hasse de \mathbb{T}_3 et \mathbb{T}_4 .

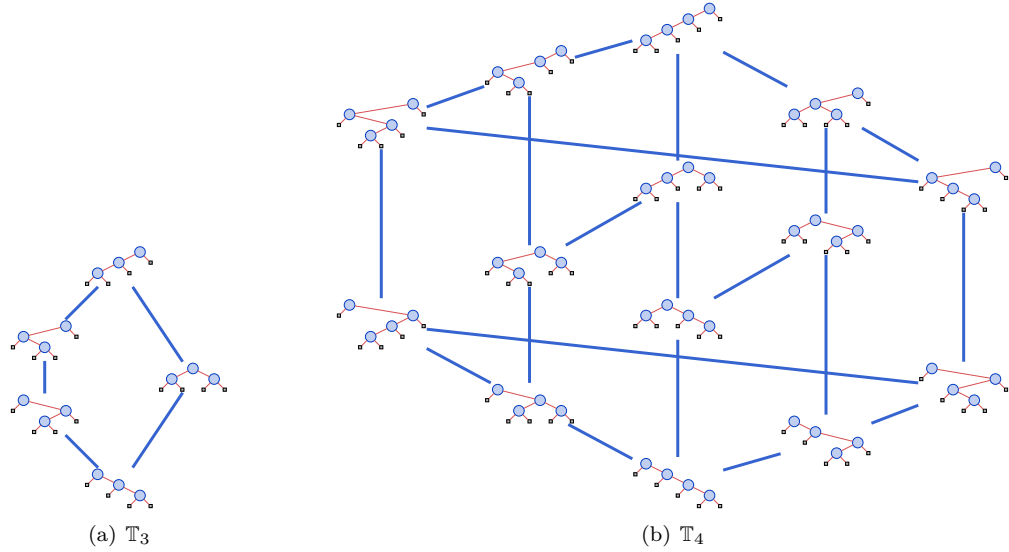


FIGURE 1.5 – Les diagrammes de Hasse des treillis de Tamari d'ordre 3 et 4.

Le plus petit élément du treillis de Tamari \mathbb{T}_n est le *peigne gauche* de taille n , c'est-à-dire l'arbre binaire à n nœuds dont chaque nœud possède un sous-arbre droit vide. De même, le plus grand élément de \mathbb{T}_n est le *peigne droit* de taille n , défini de façon similaire.

Soit T un arbre binaire étiqueté. Si pour tout nœud x de T étiqueté par une lettre b , pour tout nœud du sous-arbre gauche de x étiqueté par a , et tout nœud du sous-arbre droit de x étiqueté par c , nous avons l'inégalité $a \leq b < c$, alors T est un *arbre binaire de recherche droit*. Si nous avons à la place l'inégalité $a < b \leq c$, alors T est un *arbre binaire de recherche gauche* (voir la figure 1.6). Dans le cas où il n'y a pas de répétition dans les étiquettes de T , les notions d'arbres binaire de recherche droit et gauche coïncident.

Soit T un arbre binaire étiqueté. Si pour tous nœuds x et y de T étiquetés respectivement par a et b , x est un ancêtre de y implique $a \leq b$ (resp. $b \leq a$), alors T est un *arbre binaire croissant* (resp. *décroissant*).

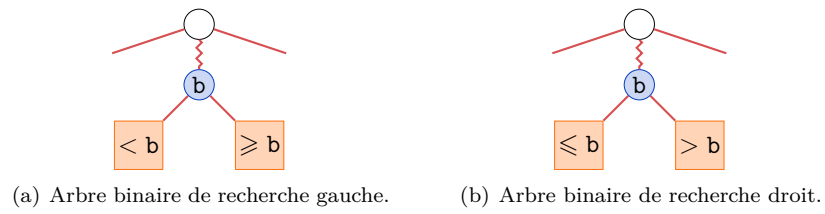


FIGURE 1.6 – Conditions sur les étiquettes des sous-arbres d'un arbre binaire de recherche.

La *forme* d'un arbre binaire étiqueté T désigne l'arbre binaire non étiqueté obtenu en oubliant les étiquettes de T .

Algèbres de Hopf combinatoires

Les algèbres de Hopf sont des structures algébriques très riches puisqu'elles sont constituées d'un espace vectoriel sur lequel est à la fois défini une structure d'algèbre et une structure de cogèbre. Dans une algèbre, on dispose d'un produit qui permet d'*assembler* des éléments, et à l'inverse, dans une cogèbre, on dispose d'un coproduit qui permet plutôt de les *désassembler*. Ces deux opérations doivent vérifier un certain nombre de compatibilités, ce qui fait que les algèbres de Hopf sont, informellement parlant, des structures algébriques extrêmement contraintes. Des références classiques sur le sujet sont entre autres [Swe69], [Abe80], et [Car06].

Lorsque l'espace vectoriel sous-jacent à une algèbre de Hopf possède des bases indexées par des objets combinatoires, son produit et son coproduit sont la plupart du temps reformulables en termes d'algorithmes combinatoires. Moyennant quelques conditions supplémentaires, un tel objet est appelé de manière heuristique *algèbre de Hopf combinatoire*, dont la définition exacte n'est à l'heure actuelle toujours pas fixée — il est intéressant de voir à ce propos [ABS03] ou [LR10] qui donnent chacun de leur côté leur propre définition de ce concept. Ce sont précisément ces objets qui occupent une place de choix dans ce mémoire, et nous proposons nous aussi une définition adaptée à notre contexte. De bonnes introductions aux algèbres de Hopf combinatoires sont [Hiv03] et [Bla10].

La littérature dans ce domaine est très abondante : citons par exemple [MR95] (voir également [DHT02]) où une algèbre de Hopf sur les permutations est introduite, mais aussi [NTT04] et [HNT08a] où des algèbres de Hopf combinatoires sur des graphes étiquetés sont construites et étudiées, ou encore [NT04] et [NT07] où les auteurs construisent une algèbre de Hopf sur les fonctions de parking et plusieurs sous-algèbres de Hopf de celle-ci dont l'une est basée sur les arbres de Schröder et une autre sur les objets catalans. Une grande quantité d'algèbres de Hopf combinatoires mettent en jeu des arbres, parmi lesquelles celle de Loday-Ronco sur les arbres binaires [LR98] (voir aussi [HNT02]), et celle de Connes-Kreimer sur les forêts d'arbres enracinés [CK98] sont sans doute les plus célèbres.

Une manière très avantageuse de voir ces algèbres de Hopf est d'interpréter ses éléments comme des polynômes — commutatifs ou non. Le produit de l'algèbre devient un simple produit polynomial, et le coproduit de la cogèbre étant obtenu par une transformation d'alphabet [KLT97] adéquate. Cette manière de voir les choses simplifie considérablement la théorie puisque la plupart des produits, définis sur les objets combinatoires eux-mêmes, peuvent être a priori extrêmement complexes, en contraste avec la simplicité du produit de polynômes offerte a posteriori par une réalisation polynomiale. De plus, la compatibilité entre le produit et le coproduit, propriété essentielle dans une algèbre de Hopf, devient immédiate. On appelle ce procédé une *réalisation polynomiale* d'une algèbre de Hopf. Plusieurs algèbres de Hopf combinatoires ont été ainsi *réalisées* : l'algèbre de Hopf des permutations [DHT02], celle de Loday-Ronco des arbres

binaires [HNT02], celle des fonctions de Parking et des arbres de Schröder [NT07], et celle de Connes-Kreimer [FNT10].

Ce chapitre préliminaire est organisé comme suit. Nous commençons en donnant dans le paragraphe 2.1 la définition de plusieurs structures algébriques : les algèbres associatives, les cogèbres, et les bigèbres, ainsi que les notions de base qui s’y rattachent. Nous terminons ce paragraphe en formalisant dans notre contexte la notion de réalisation polynomiale d’une algèbre de Hopf. Finalement, les concepts présentés sont illustrés dans le paragraphe 2.2 par des exemples. Nous passons ainsi en revue l’algèbre de Hopf de mélange et de déconcaténation, l’algèbre de Hopf de Connes-Kreimer, et l’algèbre de Hopf de Malvenuto-Reutenauer.

2.1 Définitions et propriétés de base

Nous rappelons ici les concepts généraux propres aux algèbres, cogèbres, bigèbres, et algèbres de Hopf, et définissons ensuite ce que nous entendons par réalisation polynomiale d’une algèbre de Hopf.

Dans certains cas, certaines propriétés s’expriment de manière agréable par l’intermédiaire de diagrammes commutatifs. Ainsi, il nous arrivera ponctuellement d’utiliser le langage de la théorie des catégories. Un texte de base sur le sujet comme [Pie91] est amplement suffisant dans notre contexte.

2.1.1 Algèbres combinatoires

Algèbres associatives unitaires

Définition 2.1.1. Une algèbre associative unitaire — ou simplement algèbre lorsque le contexte est clair — est un espace vectoriel \mathcal{A} muni d’une application linéaire $\cdot : \mathcal{A} \otimes \mathcal{A} \rightarrow \mathcal{A}$ appelée produit, et d’une application linéaire $u : \mathbb{K} \rightarrow \mathcal{A}$ appelée unité telles que, pour tout $x, y, z \in \mathcal{A}$ et $\lambda \in \mathbb{K}$,

$$(x \cdot y) \cdot z = x \cdot (y \cdot z), \quad (2.1.1)$$

$$u(\lambda) \cdot x = \lambda x = x \cdot u(\lambda). \quad (2.1.2)$$

La condition (2.1.1) signifie que le produit \cdot est associatif, ce qui revient à dire que le diagramme suivant est commutatif, où $I : \mathcal{A} \rightarrow \mathcal{A}$ est l’application identité, et le symbole \cdot est remplacé par le symbole p par souci de lisibilité :

$$\begin{array}{ccc} \mathcal{A} \otimes \mathcal{A} \otimes \mathcal{A} & \xrightarrow{I \otimes p} & \mathcal{A} \otimes \mathcal{A} \\ p \otimes I \downarrow & & \downarrow p \\ \mathcal{A} \otimes \mathcal{A} & \xrightarrow{p} & \mathcal{A} \end{array} \quad (2.1.3)$$

La condition (2.1.2) signifie que $u(1_{\mathbb{K}})$ est l’élément neutre du produit \cdot , ce qui revient à dire que le diagramme suivant est commutatif :

$$\begin{array}{ccccc} \mathcal{A} \otimes \mathbb{K} & \xrightarrow{I \otimes u} & \mathcal{A} \otimes \mathcal{A} & \xleftarrow{u \otimes I} & \mathbb{K} \otimes \mathcal{A} \\ & \searrow \simeq & \downarrow p & \swarrow \simeq & \\ & & \mathcal{A} & & \end{array} \quad (2.1.4)$$

Selon nos besoins, nous serons amenés à considérer des algèbres \mathcal{A} où l’unité u n’est pas donnée par une application $u : \mathbb{K} \rightarrow \mathcal{A}$ mais par un élément $1_{\mathcal{A}} \in \mathcal{A}$, qui est neutre pour

le produit. Ces deux présentations sont équivalentes puisque l'on déduit u à partir de $1_{\mathcal{A}}$ en posant $u(\lambda) := \lambda 1_{\mathcal{A}}$ pour tout $\lambda \in \mathbb{K}$, et réciproquement, on déduit $1_{\mathcal{A}}$ à partir de u en posant $1_{\mathcal{A}} := u(1_{\mathbb{K}})$.

Voyons à présent quelques définitions de base sur les algèbres. Soit $(\mathcal{A}, \cdot_{\mathcal{A}}, u_{\mathcal{A}})$ une algèbre. Un sous-espace vectoriel \mathcal{A}' de \mathcal{A} est une *sous-algèbre* de \mathcal{A} si pour tous $x, y \in \mathcal{A}'$, $x \cdot_{\mathcal{A}} y \in \mathcal{A}'$, et pour tout $\lambda \in \mathbb{K}$, $u_{\mathcal{A}}(\lambda) \in \mathcal{A}'$. Nous dirons que \mathcal{A} est *graduée* si l'espace vectoriel \mathcal{A} est gradué et

$$x \in \mathcal{A}^{(n)} \text{ et } y \in \mathcal{A}^{(m)} \quad \text{impliquent} \quad x \cdot_{\mathcal{A}} y \in \mathcal{A}^{(n+m)}. \quad (2.1.5)$$

Lorsque \mathcal{A} est graduée et que $\dim \mathcal{A}^{(0)} = 1$, \mathcal{A} est dite *connexe*. L'algèbre \mathcal{A} est *commutative* si pour tous $x, y \in \mathcal{A}$, $x \cdot_{\mathcal{A}} y = y \cdot_{\mathcal{A}} x$. Soit maintenant $(\mathcal{B}, \cdot_{\mathcal{B}}, u_{\mathcal{B}})$ une autre algèbre. Un *morphisme d'algèbre* est une application linéaire $\phi : \mathcal{A} \rightarrow \mathcal{B}$ telle que pour tous $x, y \in \mathcal{A}$ et $\lambda \in \mathbb{K}$,

$$\phi(x \cdot_{\mathcal{A}} y) = \phi(x) \cdot_{\mathcal{B}} \phi(y), \quad (2.1.6)$$

et

$$\phi(u_{\mathcal{A}}(\lambda)) = u_{\mathcal{B}}(\lambda). \quad (2.1.7)$$

Lorsque \mathcal{A} et \mathcal{B} sont graduées, on impose de plus la condition

$$x \in \mathcal{A}^{(n)} \quad \text{implique} \quad \phi(x) \in \mathcal{B}^{(n)}. \quad (2.1.8)$$

Le *produit tensoriel* de \mathcal{A} et \mathcal{B} est l'algèbre $\mathcal{A} \otimes \mathcal{B}$ avec le produit \cdot défini linéairement pour tous $x \otimes y, x' \otimes y' \in \mathcal{A} \otimes \mathcal{B}$ par

$$(x \otimes y) \cdot (x' \otimes y') := (x \cdot_{\mathcal{A}} x') \otimes (y \cdot_{\mathcal{B}} y'), \quad (2.1.9)$$

et l'unité u définie par $u := u_{\mathcal{A}} \otimes u_{\mathcal{B}}$. Un *idéal* de \mathcal{A} est un sous-espace vectoriel V de \mathcal{A} tel que, pour tout $x \in V$ et pour tout $y \in \mathcal{A}$, $x \cdot_{\mathcal{A}} y \in V$ et $y \cdot_{\mathcal{A}} x \in V$. Le *quotient* de \mathcal{A} par l'idéal V est l'algèbre \mathcal{A}/V avec le produit \cdot défini pour tous $\hat{x}, \hat{y} \in \mathcal{A}/V$ par

$$\hat{x} \cdot \hat{y} := \tau(x \cdot_{\mathcal{A}} y), \quad (2.1.10)$$

où $\tau : \mathcal{A} \rightarrow \mathcal{A}/V$ est la projection canonique et x et y sont des éléments de \mathcal{A} tels que $\tau(x) = \hat{x}$ et $\tau(y) = \hat{y}$, et l'unité u définie par $u := \tau \circ u_{\mathcal{A}}$.

L'algèbre tensorielle

Soit V un espace vectoriel de dimension potentiellement infinie, et l'espace vectoriel

$$T(V) := \mathbb{K} \oplus V \oplus V^{\otimes 2} \oplus V^{\otimes 3} \oplus \dots, \quad (2.1.11)$$

où l'espace $V^{\otimes k}$ est défini récursivement par $V^{\otimes 0} := \mathbb{K}$ et $V^{\otimes k+1} := V^{\otimes k} \otimes V$. Les éléments de $T(V)$ sont de la forme $u_1 \otimes \dots \otimes u_n$, où $n \geq 0$ et $u_i \in V$. Nous noterons ces éléments simplement par $u_1 \dots u_n$. L'espace $T(V)$ est muni d'un produit \cdot défini linéairement par

$$u_1 \dots u_n \cdot v_1 \dots v_m := u_1 \dots u_n v_1 \dots v_m, \quad (2.1.12)$$

pour tous $u_1 \dots u_n, v_1 \dots v_m \in T(V)$. Le *degré* d'un élément $x \in V^{\otimes n}$ est l'entier n . L'espace $T(V)$ admet comme unité l'unique élément de degré 0. De plus, comme le produit d'un élément de degré n avec un élément de degré m donne un élément de degré $n + m$, $T(V)$ admet une structure d'algèbre graduée et connexe : c'est l'*algèbre tensorielle* de V .

L'algèbre symétrique

L'algèbre symétrique $S(V)$ sur l'espace vectoriel V est définie par

$$S(V) := T(V)/W, \quad (2.1.13)$$

où W est le sous-espace vectoriel de $T(V)$ engendré par les éléments de la forme

$$u_1 \dots u_n - u_{\sigma_1} \dots u_{\sigma_n}, \quad (2.1.14)$$

tels que $n \geq 1$ et $\sigma \in \mathfrak{S}_n$. L'espace W est clairement un idéal de $T(V)$. Les éléments de $S(V)$ peut donc être interprétés comme des éléments $u_1 \dots u_n$ de $T(V)$ où l'ordre des u_i ne compte pas. Le produit dans l'algèbre symétrique est ainsi clairement commutatif.

Algèbres libres, bases algébriques et bases multiplicatives

Une algèbre non commutative (resp. commutative) \mathcal{A} est *libre* s'il existe un espace vectoriel V et un isomorphisme d'algèbre $\phi : \mathcal{A} \rightarrow T(V)$ (resp. $\phi : \mathcal{A} \rightarrow S(V)$).

Une base d'une algèbre \mathcal{A} est dite *multiplicative* si le produit de deux éléments de base s'exprime par un unique élément de base ayant $1_{\mathbb{K}}$ comme coefficient. De plus, un élément de base est dit *indécomposable* s'il est impossible de l'exprimer par un produit non trivial d'éléments de base.

Un ensemble G d'éléments de \mathcal{A} est un ensemble de *générateurs algébriques* de \mathcal{A} si \mathcal{A} est la plus petite algèbre qui contient G . De plus, lorsque \mathcal{A} est libre sur G , G est une *base algébrique* de \mathcal{A} . Dans ce cas, en posant $\mathcal{S}_G(t)$ la série génératrice de G définie par

$$\mathcal{S}_G(t) := \sum_{g \in G} t^{|g|}, \quad (2.1.15)$$

où $|g|$ désigne le degré de g , alors la série de Hilbert de \mathcal{A} vérifie

$$F_{\mathcal{A}}(t) = \frac{1}{1 - \mathcal{S}(t)}. \quad (2.1.16)$$

Algèbres combinatoires

Dans ce mémoire, nous considérerons plus particulièrement la sous-classe des *algèbres combinatoires* :

Définition 2.1.2. Une algèbre combinatoire est une algèbre dont l'espace vectoriel sous-jacent est combinatoire.

Notons que d'après la définition 1.1.5 du chapitre 1, tout espace vectoriel combinatoire est gradué, et la dimension de sa composante homogène de degré 0 et 1. Ainsi, une algèbre combinatoire est en particulier une algèbre graduée, connexe, et ses composantes homogènes sont de dimension finie.

Lorsque (\mathcal{A}, \cdot, u) est une algèbre combinatoire, étant donné qu'elle est graduée et connexe, la seule façon de définir l'unité u est de poser linéairement $u(1_{\mathbb{K}}) := 1$ où 1 est l'unique élément de \mathcal{A} de degré zéro. Ainsi, nous ne mentionnerons plus dans la suite l'unité des algèbres combinatoires, seul l'espace vectoriel sous-jacent et le produit le nécessitent.

La plupart des algèbres combinatoires s'implantent facilement, par exemple en Sage [S⁺11]. Il suffit en effet de spécifier la classe combinatoire sous-jacente et de programmer le produit de deux éléments de base. Le système étend automatiquement le produit par linéarité. Un bon nombre d'implantations ont été réalisées dans ce travail pour des expérimentations.

Exemples et notations

Donnons à présent des exemples d'algèbres, dans le but de fixer quelques notations :

1. l'espace vectoriel des polynômes commutatifs sur A , noté $\mathbb{K}[A]$, avec le produit de polynômes usuel, est une algèbre. Celle-ci est de plus commutative, graduée, et connexe. Les éléments de base de cette algèbre sont les mots commutatifs sur A , *i.e.*, les mots où l'ordre des lettres ne compte pas.
2. L'espace vectoriel des polynômes non commutatifs sur A , noté $\mathbb{K}\langle A \rangle$, avec le produit non commutatif de polynômes, est une algèbre. Puisque A possède plus de deux éléments — A est un alphabet infini — cette algèbre n'est pas commutative. Elle est cependant graduée et connexe, et ses éléments de base sont les mots sur A .
3. Ces deux algèbres possèdent respectivement des analogues notés $\mathbb{K}[[A]]$ et $\mathbb{K}\langle\langle A \rangle\rangle$ dont les éléments ne sont plus des polynômes mais des séries sur A , respectivement commutatives et non commutatives.
4. Si M est un monoïde, on note $\mathbb{K}[M] := \text{Vect}(M)$ l'*algèbre du monoïde* M . Les éléments de $\mathbb{K}[M]$ sont des sommes formelles d'éléments de M à coefficients dans \mathbb{K} , son produit est le produit de M étendu par linéarité, et son unité est l'élément neutre de M .

2.1.2 Cogèbres combinatoires

Cogèbres coassociatives counitaires

Définition 2.1.3. Une cogèbre coassociative counitaire — ou simplement cogèbre lorsque le contexte est clair — est un espace vectoriel \mathcal{C} muni d'une application linéaire $\Delta : \mathcal{C} \rightarrow \mathcal{C} \otimes \mathcal{C}$ appelée coproduit, et d'une application linéaire $c : \mathcal{C} \rightarrow \mathbb{K}$ appelée counité telles que, pour tout $x \in \mathcal{C}$,

$$(\Delta \otimes I)\Delta(x) = (I \otimes \Delta)\Delta(x), \quad (2.1.17)$$

$$(c \otimes I)\Delta(x) = 1_{\mathbb{K}} \otimes x \quad \text{et} \quad (I \otimes c)\Delta(x) = x \otimes 1_{\mathbb{K}}, \quad (2.1.18)$$

où $I : \mathcal{C} \rightarrow \mathcal{C}$ est l'application identité.

Le coproduit $\Delta(x)$ d'un élément x de \mathcal{C} s'écrit comme une somme finie de tenseurs et est de ce fait de la forme

$$\Delta(x) = \sum_i x_i^L \otimes x_i^R. \quad (2.1.19)$$

Il est avantageux dans bien des cas de recourir à la notation de Sweedler qui résume (2.1.19) en

$$\Delta(x) = \sum x^L \otimes x^R. \quad (2.1.20)$$

La condition (2.1.17) signifie que le coproduit Δ est *coassociatif*, ce qui est équivalent à dire que le diagramme suivant est commutatif :

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{\Delta} & \mathcal{C} \otimes \mathcal{C} \\ \Delta \downarrow & & \downarrow I \otimes \Delta \\ \mathcal{C} \otimes \mathcal{C} & \xrightarrow{\Delta \otimes I} & \mathcal{C} \otimes \mathcal{C} \otimes \mathcal{C} \end{array} \quad (2.1.21)$$

De même, la condition (2.1.18) est encodée par le diagramme commutatif suivant :

$$\begin{array}{ccccc} \mathcal{C} \otimes \mathbb{K} & \xleftarrow{I \otimes c} & \mathcal{C} \otimes \mathcal{C} & \xrightarrow{c \otimes I} & \mathbb{K} \otimes \mathcal{C} \\ & \nwarrow \simeq & \uparrow \Delta & \nearrow \simeq & \\ & & \mathcal{C} & & \end{array} \quad (2.1.22)$$

Remarquons que la notion de cogèbre est en un certain sens duale de la notion d'algèbre. En effet, on obtient le diagramme (2.1.21) en inversant le sens des flèches du diagramme (2.1.3). Il en est de même pour le diagramme (2.1.22) qui peut être obtenu de la même façon à partir du diagramme (2.1.4). Ainsi, les notions définies dans le paragraphe 2.1.1 sont adaptables aux cogèbres. Reformulons-les tout de même.

Soit $(\mathcal{C}, \Delta_{\mathcal{C}}, c_{\mathcal{C}})$ une cogèbre. Un élément $x \in \mathcal{C}$ est dit *primitif* si $\Delta_{\mathcal{C}}(x) = 1_{\mathbb{K}} \otimes x + x \otimes 1_{\mathbb{K}}$. Il est dit de *type groupe* si $\Delta_{\mathcal{C}}(x) = x \otimes x$. Un sous-espace vectoriel \mathcal{C}' de \mathcal{C} est une *sous-cogèbre* de \mathcal{C} si pour tout $x \in \mathcal{C}'$, $\Delta(x) \in \mathcal{C}' \otimes \mathcal{C}'$. Nous dirons que \mathcal{C} est *cograduée* si l'espace vectoriel \mathcal{C} est gradué et

$$x \in \mathcal{C}^{(n)} \quad \text{implique} \quad \Delta_{\mathcal{C}}(x) \in \bigoplus_{i+j=n} \mathcal{C}^{(i)} \otimes \mathcal{C}^{(j)}. \quad (2.1.23)$$

Lorsque \mathcal{C} est cograduée et que $\dim \mathcal{C}^{(0)} = 1$, \mathcal{C} est dite *coconnexe*. Notons $\omega : \mathcal{C} \otimes \mathcal{C} \rightarrow \mathcal{C} \otimes \mathcal{C}$ l'application linéaire définie pour tout $x \otimes y \in \mathcal{C} \otimes \mathcal{C}$ par $\omega(x \otimes y) := y \otimes x$. La cogèbre \mathcal{C} est *cocommutative* si pour tous $x \in \mathcal{C}$, $\Delta_{\mathcal{C}}(x) = \omega(\Delta_{\mathcal{C}}(x))$. Soit maintenant $(\mathcal{D}, \Delta_{\mathcal{D}}, c_{\mathcal{D}})$ une autre cogèbre. Un *morphisme de cogèbre* est une application linéaire $\phi : \mathcal{C} \rightarrow \mathcal{D}$ telle que pour tout $x \in \mathcal{C}$,

$$(\phi \otimes \phi)\Delta_{\mathcal{C}}(x) = \Delta_{\mathcal{D}}(\phi(x)), \quad (2.1.24)$$

et

$$c_{\mathcal{C}}(x) = c_{\mathcal{D}}(\phi(x)). \quad (2.1.25)$$

Lorsque \mathcal{C} et \mathcal{D} sont cograduées, on impose de plus la condition

$$x \in \mathcal{C}^{(n)} \quad \text{implique} \quad \phi(x) \in \mathcal{D}^{(n)}. \quad (2.1.26)$$

Le *produit tensoriel* de \mathcal{C} et \mathcal{D} est la cogèbre $\mathcal{C} \otimes \mathcal{D}$ avec le coproduit Δ défini linéairement pour tout $x \otimes y \in \mathcal{C} \otimes \mathcal{D}$ par

$$\Delta(x \otimes y) := \sum (x^L \otimes y^L) \otimes (x^R \otimes y^R), \quad (2.1.27)$$

où $\Delta_{\mathcal{C}}(x) = \sum x^L \otimes x^R$ et $\Delta_{\mathcal{D}}(y) = \sum y^L \otimes y^R$, et la counité c définie par $c := c_{\mathcal{C}} \otimes c_{\mathcal{D}}$. Un *coidéal* de \mathcal{C} est un sous-espace vectoriel V de \mathcal{C} tel que, pour tout $x \in V$, $\Delta_{\mathcal{C}}(x) \in V \otimes \mathcal{C} + \mathcal{C} \otimes V$ et $V \subseteq \ker c$. Le *quotient* de \mathcal{C} par le coidéal V est la cogèbre \mathcal{C}/V avec le coproduit Δ défini pour tous $\hat{x} \in \mathcal{C}/V$ par

$$\Delta(\hat{x}) := (\tau \otimes \tau)\Delta_{\mathcal{C}}(x), \quad (2.1.28)$$

où $\tau : \mathcal{C} \rightarrow \mathcal{C}/V$ est la projection canonique et x est un élément de \mathcal{C} tel que $\tau(x) = \hat{x}$, et la counité c définie par $c(\hat{x}) := c_{\mathcal{C}}(x)$.

La cogèbre tensorielle et cogèbres libres

Soit V un espace vectoriel potentiellement infini. Munissons l'espace vectoriel $T(V)$ d'un coproduit Δ défini linéairement pour tout $u_1 \dots u_n \in T(V)$ par

$$\Delta(u_1 \dots u_n) := \sum_{0 \leq i \leq n} u_1 \dots u_i \otimes u_{i+1} \dots u_n, \quad (2.1.29)$$

et d'une counité $c : T(V) \rightarrow \mathbb{K}$ définie linéairement par $c(x) = \delta_{x,1}$ où 1 est l'élément de degré 0 de $T(V)$. Puisque la somme des degrés de u et de v est n , où $u \otimes v$ est un tenseur qui apparaît dans un coproduit d'un élément de degré n , $T(V)$ admet une structure de cogèbre cograduée coconnexe : c'est la *cogèbre tensorielle*. Une cogèbre \mathcal{C} est dite *colibre* s'il existe un isomorphisme de cogèbre $\phi : \mathcal{C} \rightarrow T(V)$ où V est l'espace vectoriel des éléments primitifs de \mathcal{C} .

Cogèbres combinatoires

Dans ce mémoire, nous considérerons plus particulièrement la sous-classe des *cogèbres combinatoires* :

Définition 2.1.4. *Une cogèbre combinatoire est une cogèbre dont l'espace vectoriel sous-jacent est combinatoire.*

Comme dans le cas des algèbres combinatoires, toute cogèbre combinatoire est cograduée, coconnexe, et ses composantes homogènes sont de dimension finie.

Dans ce cas aussi, lorsque (\mathcal{C}, Δ, c) est une cogèbre combinatoire, étant donné qu'elle est cograduée et coconnexe, la seule façon de définir la counité c est de poser linéairement $c(x) := \delta_{x,1}$ où 1 est l'unique élément de \mathcal{C} de degré zéro. Ainsi, nous ne mentionnerons plus dans la suite la counité des cogèbres combinatoires, seul l'espace vectoriel sous-jacent et le coproduit le nécessitent.

La plupart des cogèbres combinatoires s'implantent facilement, par exemple en Sage [S⁺11]. Il suffit en effet de spécifier la classe combinatoire sous-jacente et de programmer le coproduit d'un élément de base. Le système étend automatiquement le coproduit par linéarité. Un bon nombre d'implantations ont été réalisées dans ce travail pour des expérimentations.

2.1.3 Bigèbres et algèbres de Hopf combinatoires

Bigèbres

Définition 2.1.5. *Une bigèbre \mathcal{B} est un espace vectoriel qui possède à la fois une structure d'algèbre (\mathcal{B}, \cdot, u) et une structure de cogèbre (\mathcal{B}, Δ, c) telles que Δ et c sont des morphismes d'algèbre, ou de manière équivalente, \cdot et u des morphismes de cogèbre.*

En notant $I : \mathcal{B} \rightarrow \mathcal{B}$ l'application identité, le fait que Δ et c soient des morphismes d'algèbre, et le fait que \cdot et u soient des morphismes de cogèbre sont équivalents aux relations suivantes. Pour tous $x, y \in \mathcal{B}$,

$$\Delta(x \cdot y) = (\cdot \otimes \cdot)(I \otimes \omega \otimes I)(\Delta(x) \otimes \Delta(y)), \quad (2.1.30)$$

$$c(x \cdot y) = c(x)c(y), \quad (2.1.31)$$

$$\Delta(u(1_{\mathbb{K}})) = u(1_{\mathbb{K}}) \otimes u(1_{\mathbb{K}}), \quad (2.1.32)$$

$$c(u(1_{\mathbb{K}})) = 1_{\mathbb{K}}, \quad (2.1.33)$$

où, rappelons-le, ω est l'application linéaire définie pour tout $x \otimes y \in \mathcal{B} \otimes \mathcal{B}$ par $\omega(x \otimes y) := y \otimes x$. Ces relations sont traduites respectivement par les diagrammes commutatifs suivants :

$$\begin{array}{ccccc} \mathcal{B} \otimes \mathcal{B} & \xrightarrow{p} & \mathcal{B} & \xrightarrow{\Delta} & \mathcal{B} \otimes \mathcal{B} \\ \Delta \otimes \Delta \downarrow & & & & \uparrow p \otimes p \\ \mathcal{B} \otimes \mathcal{B} \otimes \mathcal{B} \otimes \mathcal{B} & \xrightarrow{I \otimes \omega \otimes I} & \mathcal{B} \otimes \mathcal{B} \otimes \mathcal{B} \otimes \mathcal{B} & & \end{array} \quad (2.1.34)$$

$$\begin{array}{ccc} & \mathcal{B} & \\ p \nearrow & & \searrow c \\ \mathcal{B} \otimes \mathcal{B} & \xrightarrow{c \otimes c} & \mathbb{K} \otimes \mathbb{K} \simeq \mathbb{K} \end{array} \quad (2.1.35)$$

$$\begin{array}{ccc}
& \mathcal{B} & \\
\Delta \swarrow & & \searrow u \\
\mathcal{B} \otimes \mathcal{B} & \xleftarrow{u \otimes u} & \mathbb{K} \otimes \mathbb{K} \simeq \mathbb{K}
\end{array} \tag{2.1.36}$$

$$\begin{array}{ccc}
& \mathcal{B} & \\
u \swarrow & & \searrow c \\
\mathbb{K} & \xrightarrow{I} & \mathbb{K}
\end{array} \tag{2.1.37}$$

où l'application I dans le dernier diagramme est l'identité sur \mathbb{K} .

Voyons à présent quelques définitions de base à propos des bigèbres. Soit $(\mathcal{B}, \cdot_{\mathcal{B}}, u_{\mathcal{B}}, \Delta_{\mathcal{B}}, c_{\mathcal{B}})$ une bigèbre. Un sous-espace vectoriel \mathcal{B}' de \mathcal{B} est une *sous-bigèbre* de \mathcal{B} si \mathcal{B}' est à la fois une sous-algèbre et une sous-cogèbre de \mathcal{B} . Nous dirons que \mathcal{B} est *graduée* si \mathcal{B} est à la fois graduée en tant qu'algèbre et cograduée en tant que cogèbre. De plus, quand \mathcal{B} est graduée et que \mathcal{B} est connexe en tant qu'algèbre et coconnexe en tant que cogèbre, \mathcal{B} est qualifiée de *connexe*. Soit maintenant $(\mathcal{C}, \cdot_{\mathcal{C}}, u_{\mathcal{C}}, \Delta_{\mathcal{C}}, c_{\mathcal{C}})$ une autre bigèbre. Un *morphisme de bigèbre* est une application linéaire $\phi : \mathcal{B} \rightarrow \mathcal{C}$ qui est à la fois un morphisme d'algèbre et un morphisme de cogèbre. Le *produit tensoriel* de \mathcal{B} et \mathcal{C} est la bigèbre $\mathcal{B} \otimes \mathcal{C} =: \mathcal{D}$, où \mathcal{D} est en tant qu'algèbre (resp. en tant que cogèbre) le produit tensoriel des algèbres (resp. des cogèbres) \mathcal{B} et \mathcal{C} . Un *idéal de bigèbre* V de \mathcal{B} est à la fois un idéal de \mathcal{B} en tant qu'algèbre, et un coidéal de \mathcal{B} en tant que cogèbre. Le sous-espace $\ker c_{\mathcal{B}}$ forme un idéal particulier de \mathcal{B} : on l'appelle *idéal d'augmentation* de \mathcal{B} et on le note \mathcal{B}^+ . Le *quotient* de \mathcal{B} par l'idéal V est la bigèbre $\mathcal{B}/_V$, qui est en tant qu'algèbre (resp. que cogèbre) le quotient de l'algèbre \mathcal{B} (resp. de la cogèbre \mathcal{B}) par l'idéal V (resp. le coidéal V).

Bases du côté gauche

En suivant la terminologie de [LR09], une base d'une bigèbre \mathcal{B} est *du côté gauche* si elle est multiplicative, et pour tout élément indécomposable x , en utilisant la notation de Sweedler, nous avons

$$\Delta(x) = \sum x^L \otimes x^R, \tag{2.1.38}$$

où les x^L sont des éléments indécomposables.

Bigèbres combinatoires

Dans ce mémoire, nous considérerons plus particulièrement la sous-classe des *bigèbres combinatoires* :

Définition 2.1.6. Une bigèbre combinatoire est une bigèbre dont l'espace vectoriel sous-jacent est combinatoire.

Comme dans le cas des algèbres et des cogèbres combinatoires, toute bigèbre combinatoire est graduée, connexe, et ses composantes homogènes sont de dimension finie.

Comme nous l'avons mentionné dans les paragraphes 2.1.1 et 2.1.2, toute algèbre (resp. cogèbre) combinatoire est entièrement spécifiée par la donnée de son espace vectoriel et de son produit (resp. coproduit). Ainsi, une bigèbre combinatoire est entièrement définie par un triplet $(\mathcal{B}, \cdot, \Delta)$.

Étant donné que toutes les composantes homogènes d'une bigèbre combinatoire sont de dimensions finies, la définition suivante est valide :

Définition 2.1.7. Soit $(\mathcal{B}, \cdot, \Delta)$ une bigèbre combinatoire. Le dual gradué de \mathcal{B} est la bigèbre combinatoire $(\mathcal{B}^*, \cdot^*, \Delta^*)$ où \mathcal{B}^* est le dual gradué de \mathcal{B} en tant qu'espace vectoriel, et les applications \cdot^* et Δ^* sont respectivement les applications transposées de Δ et \cdot . En d'autres termes, le produit \cdot^* et le coproduit Δ^* vérifient

$$\langle x \cdot y, z \rangle = \langle x \otimes y, \Delta^*(z) \rangle \quad \text{pour tous } x \in \mathcal{B}^{(n)}, y \in \mathcal{B}^{(m)}, \text{ et } z \in \mathcal{B}^{*(n+m)}, \quad (2.1.39)$$

$$\langle \Delta(x), y \otimes z \rangle = \langle x, y \cdot^* z \rangle \quad \text{pour tous } x \in \mathcal{B}^{(n+m)}, y \in \mathcal{B}^{*(n)}, \text{ et } z \in \mathcal{B}^{*(m)}, \quad (2.1.40)$$

où $\langle -, - \rangle : \mathcal{B} \otimes \mathcal{B}^* \rightarrow \mathbb{K}$ est le crochet de dualité.

Dans la suite, nous parlerons simplement de *dual* pour le dual gradué d'une bigèbre combinatoire. Une bigèbre \mathcal{B} est *autoduale* s'il existe un isomorphisme de bigèbre $\phi : \mathcal{B} \rightarrow \mathcal{B}^*$.

Produit de convolution et algèbres de Hopf

Soit $(\mathcal{B}, \cdot, u, \Delta, c)$ une bigèbre — non nécessairement combinatoire. Notons \mathcal{E} l'espace vectoriel des applications linéaires de \mathcal{B} dans \mathcal{B} . Nous pouvons munir \mathcal{E} d'un produit $*$ défini pour tous $f, g \in \mathcal{E}$ par :

$$f * g := \cdot \circ (f \otimes g) \circ \Delta. \quad (2.1.41)$$

De manière équivalente, en utilisant la notation de Sweedler, on a pour tout $x \in \mathcal{B}$,

$$(f * g)(x) = \sum f(x^L) \cdot g(x^R). \quad (2.1.42)$$

L'associativité de \cdot et la coassociativité de Δ impliquent que $*$ est associatif. Ce produit est appelé *produit de convolution*. D'autre part, on vérifie facilement que l'élément $u \circ c$ de \mathcal{E} est l'élément neutre du produit de convolution. La structure $(\mathcal{E}, *, u \circ c)$ est donc une algèbre.

Nous pouvons maintenant définir la notion d'algèbre de Hopf :

Définition 2.1.8. Une algèbre de Hopf \mathcal{H} est une bigèbre $(\mathcal{H}, \cdot, u, \Delta, c)$ munie d'une application linéaire $S : \mathcal{H} \rightarrow \mathcal{H}$ appelée antipode. Cette dernière est l'inverse de l'identité $I : \mathcal{H} \rightarrow \mathcal{H}$ pour le produit de convolution.

En d'autres termes, l'antipode S d'une algèbre de Hopf \mathcal{H} vérifie

$$S * I = I * S = u \circ c, \quad (2.1.43)$$

ce qui équivaut à dire que S rend le diagramme suivant commutatif :

$$\begin{array}{ccccc} & & \mathcal{H} \otimes \mathcal{H} & \xrightarrow{I \otimes S} & \mathcal{H} \otimes \mathcal{H} \\ & \nearrow \Delta & & & \searrow p \\ \mathcal{H} & \xrightarrow{c} & \mathbb{K} & \xrightarrow{u} & \mathcal{H} \\ & \searrow \Delta & & & \nearrow p \\ & & \mathcal{H} \otimes \mathcal{H} & \xrightarrow{S \otimes I} & \mathcal{H} \otimes \mathcal{H} \end{array} \quad (2.1.44)$$

Citons quelques propriétés notables des antipodes. Soit \mathcal{H} une algèbre de Hopf et S son antipode. Alors, S est la seule application linéaire qui vérifie (2.1.43). Toute bigèbre admet donc au plus une structure d'algèbre de Hopf. De plus, S est un antihomomorphisme de Hopf de \mathcal{H} , i.e.,

$$S(x \cdot y) = S(y) \cdot S(x), \quad (2.1.45)$$

pour tous $x, y \in \mathcal{H}$. Enfin, si \mathcal{H} est commutative ou cocommutative, alors S est une involution, i.e., $S \circ S$ est l'application identité.

Algèbres de Hopf combinatoires

Nous pouvons maintenant poser la définition fondamentale de ce paragraphe :

Définition 2.1.9. *Une algèbre de Hopf combinatoire est une bigèbre combinatoire.*

Remarquons que la définition 2.1.9 est consistante car toute bigèbre combinatoire $(\mathcal{B}, \cdot, \Delta)$ admet un antipode S , ce qui fait d'elle automatiquement une algèbre de Hopf. En effet, en se basant sur le fait que \mathcal{B} est combinatoire, cette dernière est graduée et connexe, et ceci implique que S se calcule degré par degré selon la formule suivante, que l'on écrit à l'aide de la notation de Sweedler :

$$S(x) = \begin{cases} x & \text{si } x \in \mathcal{B}^{(0)}, \\ - \sum_{x^L \neq x} S(x^L) \cdot x^R & \text{sinon.} \end{cases} \quad (2.1.46)$$

Ainsi, nous ne détaillerons presque jamais l'antipode des algèbres de Hopf combinatoires que nous rencontrerons dans ce mémoire, leur existence est assurée par (2.1.46). Leur étude, souvent intéressante du point de vue combinatoire, ne rentre néanmoins pas dans le cadre de notre travail.

La plupart des algèbres de Hopf combinatoires s'implantent facilement, par exemple en Sage [S⁺11]. Il suffit en effet de spécifier la classe combinatoire sous-jacente et de programmer le produit de deux éléments de base et le coproduit d'un élément de base. Le système étend automatiquement le produit et le coproduit par linéarité. Pour réaliser des expérimentations, un grand nombre d'algèbres de Hopf combinatoires ont été implantées.

2.1.4 Réalisations polynomiales d'algèbres de Hopf

Nous décrivons dans ce paragraphe une parcelle, suffisante pour le contenu des chapitres qui suivent, de la théorie des réalisations polynomiales d'algèbres de Hopf.

L'algèbre des polynômes non commutatifs

L'algèbre des polynômes non commutatifs $\mathbb{K}\langle A \rangle$ sur l'alphabet de variables A peut être vue comme l'algèbre tensorielle sur l'espace vectoriel $\text{Vect}(A)$. Pour cette raison, $\mathbb{K}\langle A \rangle$ est libre, et nous l'appelons *algèbre associative libre* sur A . Notons cependant que $\mathbb{K}\langle A \rangle$ n'est pas une algèbre combinatoire au sens de la définition 2.1.2 puisque, comme A est infini, il existe une infinité de mots sur A de taille $n \geq 1$, impliquant que les dimensions des composantes homogènes de degrés $n \geq 1$ de $\mathbb{K}\langle A \rangle$ sont infinies. Cette algèbre joue malgré tout un rôle central dans la théorie des réalisations polynomiales d'algèbres de Hopf.

Réalisations polynomiales d'algèbres

Définition 2.1.10. *Soit \mathcal{A} une algèbre. Une application linéaire $r_A : \mathcal{A} \rightarrow \mathbb{K}\langle A \rangle$ est une réalisation polynomiale d'algèbre de \mathcal{A} si, pour tout alphabet A infini, r_A est un morphisme d'algèbre injectif.*

En d'autres termes, une réalisation polynomiale d'une algèbre \mathcal{A} permet d'encoder ses éléments par des polynômes de $\mathbb{K}\langle A \rangle$ où A est un alphabet infini quelconque. Le fait que r_A soit injectif assure que cet encodage est sans perte d'information. De plus, le fait que r_A soit un morphisme d'algèbre assure qu'il est possible de calculer le produit dans \mathcal{A} de deux éléments x et y en considérant le produit de leurs polynômes réalisateurs $r_A(x)$ et $r_A(y)$. Le polynôme $r_A(x) \cdot r_A(y)$ est en effet le polynôme réalisateur $r_A(x \cdot y)$ de l'élément $x \cdot y$.

Sommes ordinales de \prec -alphabets et algèbres associées

Tel qu'il est défini pour le moment, l'alphabet A est muni d'une relation d'ordre totale \leq . Cependant, nous serons amenés dans la suite à considérer que A est plus généralement muni d'une structure supplémentaire. Par structure supplémentaire, nous entendons une relation binaire \prec sur ses éléments. Dans ce cas, nous dirons que A est un \prec -alphabet.

Soient (A, \prec_A) et (B, \prec_B) deux \prec -alphabets. Nous notons $A \oplus B$ la *somme ordinale* des alphabets A et B , c'est-à-dire, l'ensemble $A \uplus B$ muni de la relation \prec où l'on pose, pour tous $x, y \in A \uplus B$,

$$x \prec y \quad \text{si et seulement si} \quad \begin{array}{l} x \in A, y \in A \text{ et } x \prec_A y, \\ \text{ou} \\ x \in B, y \in B \text{ et } x \prec_B y, \\ \text{ou} \\ x \in A \text{ et } y \in B. \end{array} \quad (2.1.47)$$

L'algèbre $\mathbb{K}\langle A \oplus B \rangle$ sur l'alphabet constitué de la somme ordinale de A et de B est l'algèbre des polynômes non commutatifs sur l'alphabet $A \uplus B$ où les lettres de A et de B commutent entre elles. En d'autres termes, cette algèbre est un quotient de l'algèbre associative libre sur l'alphabet $A \uplus B$, et nous avons

$$\mathbb{K}\langle A \oplus B \rangle = \mathbb{K}\langle A \uplus B \rangle / V, \quad (2.1.48)$$

où V est le sous-espace vectoriel de $\mathbb{K}\langle A \uplus B \rangle$ défini par

$$V := \text{Vect} \left(\{u - v : u, v \in (A \uplus B)^+, u|_A = v|_A, \text{ et } u|_B = v|_B\} \right). \quad (2.1.49)$$

Tout élément de base x de $\mathbb{K}\langle A \oplus B \rangle$ peut ainsi être écrit sous la forme

$$x = u \cdot v, \quad (2.1.50)$$

où u est un mot sur l'alphabet A , et v un mot sur l'alphabet B . L'élément x admet ainsi l'écriture

$$x = u \otimes v, \quad (2.1.51)$$

ce qui montre que x peut être vu comme un élément de $\mathbb{K}\langle A \rangle \otimes \mathbb{K}\langle B \rangle$ et que

$$\mathbb{K}\langle A \oplus B \rangle \simeq \mathbb{K}\langle A \rangle \otimes \mathbb{K}\langle B \rangle. \quad (2.1.52)$$

Réalisations polynomiales de cogèbres

Définition 2.1.11. Soit \mathcal{C} une cogèbre. Une application linéaire $r_A : \mathcal{C} \rightarrow \mathbb{K}\langle A \rangle$ est une réalisation polynomiale de cogèbre de \mathcal{C} si, pour tout alphabet A infini muni d'une relation \prec , r_A est injective, et de plus, pour tout alphabet B infini muni d'une relation \prec , on a

$$(r_A \otimes r_B) \Delta(x) = r_{A \oplus B}(x). \quad (2.1.53)$$

En d'autres termes, comme dans le cas précédent des réalisations polynomiales d'algèbres, une réalisation polynomiale d'une cogèbre \mathcal{C} permet d'encoder ses éléments par des polynômes de $\mathbb{K}\langle A \rangle$. Le fait que r_A soit injectif assure que cet encodage est sans perte d'information. En outre, (2.1.53) permet de calculer le coproduit d'un élément x de \mathcal{C} par la technique du *doublément d'alphabet* sur le polynôme réalisateur de x (voir aussi à ce propos [Hiv03]). Il devient en effet possible de calculer le coproduit de x en considérant le polynôme réalisateur de x sur l'alphabet $A \oplus A$. Grâce à l'identification (2.1.52), le polynôme $r_{A \oplus A}(x)$ est en effet le polynôme réalisateur $(r_A \otimes r_A) \Delta(x)$ de l'élément $\Delta(x)$.

Réalisations polynomiales d'algèbres de Hopf

Définition 2.1.12. Soit \mathcal{H} une algèbre de Hopf. Une réalisation polynomiale d'algèbre de Hopf de \mathcal{H} est une application linéaire $r_A : \mathcal{H} \rightarrow \mathbb{K}\langle A \rangle$ qui est à la fois une réalisation polynomiale de \mathcal{H} en tant qu'algèbre et une réalisation polynomiale de \mathcal{H} en tant que cogèbre.

Plusieurs réalisations polynomiales d'algèbres de Hopf furent découvertes récemment et chacune d'elle nécessite un \prec -alphabet particulier. Par exemple, l'algèbre de Hopf des permutations **FQSym** [DHT02] (voir le paragraphe 2.2.3), l'algèbre de Hopf **WQSym** [NT06] basée sur les mots tassés et l'algèbre de Hopf **PQSym** [NT07] basée sur des fonctions de parking ainsi que certaines de leurs sous-algèbres de Hopf sont réalisées en utilisant un alphabet de variables non commutatives où \prec est une relation d'ordre totale. La réalisation polynomiale de l'algèbre de Hopf de Connes-Kreimer [FNT10] nécessite quant à elle un alphabet de variables non commutatives et bi-indexées, où \prec n'est dans ce cas pas un ordre mais seulement une relation binaire.

2.2 Exemples d'algèbres de Hopf combinatoires

Il est maintenant temps d'illustrer les concepts rappelés dans le paragraphe précédent. Nous proposons ici trois exemples d'algèbres de Hopf combinatoires. Le premier est une algèbre de Hopf bien connue sur les mots [Reu93]. Le second porte sur l'algèbre de Connes-Kreimer, introduite en [CK98] et [Kre98] dans le contexte de la renormalisation dans la théorie quantique des champs. Notre dernier exemple traite d'une algèbre de Hopf basée sur les permutations : l'algèbre de Hopf de Malvenuto-Reutenauer [MR95]. Nous poserons ici les notations principales à son sujet, en vue d'y faire référence au chapitre 5.

2.2.1 L'algèbre de Hopf de mélange et de déconcaténation

L'algèbre de mélange

Soit L un alphabet fini et non vide. L'ensemble L^* des mots finis sur L où la taille d'un élément est sa longueur est une classe combinatoire au sens de la définition 1.1.1 du chapitre 1. Ainsi, et comme il existe exactement un mot de longueur 0, l'espace vectoriel $\text{Vect}(L^*) =: \mathcal{M}$ est un espace vectoriel combinatoire au sens de la définition 1.1.5 et l'ensemble L^* forme une base. En munissant \mathcal{M} du produit de mélange \sqcup (sa définition est donnée en (1.2.6) dans le chapitre 1), on obtient une algèbre combinatoire : l'algèbre de mélange.

La cogèbre de déconcaténation

Sur le même espace \mathcal{M} nous pouvons définir linéairement un coproduit Δ par

$$\Delta(u) := \sum_{u=v \cdot w} v \otimes w, \quad (2.2.1)$$

pour tout mot $u \in L^*$. Ce coproduit est connu sous le nom de *coproduit de déconcaténation* [LR06]. Nous obtenons ainsi une cogèbre combinatoire : la *cogèbre de déconcaténation*. Remarquons au passage que (\mathcal{M}, Δ, c) est une *cogèbre colibre* (voir [LR06]).

Voici un exemple de coproduit avec $L := \{a, b, c\}$:

$$\Delta(\text{bacc}) = 1 \otimes \text{bacc} + b \otimes \text{acc} + ba \otimes cc + bac \otimes c + \text{bacc} \otimes 1, \quad (2.2.2)$$

où le mot vide est noté 1.

L'algèbre de Hopf de mélange et de déconcaténation

Le produit \sqcup et le coproduit Δ définis sur \mathcal{M} vérifient les conditions de la définition 2.1.5, et de ce fait, \mathcal{M} est une bigèbre combinatoire [Mal93]. De plus, comme nous l'avons mentionné dans le paragraphe 2.1.3, toute bigèbre combinatoire peut être munie d'un antipode S défini selon (2.1.46), ce qui fait d'elle une algèbre de Hopf combinatoire. On peut montrer par récurrence sur la longueur des mots de L^* que S est défini linéairement par

$$S(u) := (-1_{\mathbb{K}})^{|u|} u^{\sim}, \quad (2.2.3)$$

pour tout $u \in L^*$. Notons finalement que \mathcal{M} est commutative et non cocommutative, et que sa série de Hilbert est

$$F_{\mathcal{M}}(t) = \sum_{n \geq 0} (\#L)^n t^n. \quad (2.2.4)$$

Nous appelons \mathcal{M} l'*algèbre de Hopf de mélange et de déconcaténation*.

L'algèbre de Hopf duale

Intéressons-nous à présent à l'algèbre de Hopf duale \mathcal{M}^* . Le crochet de dualité $\langle -, - \rangle : \mathcal{M} \otimes \mathcal{M}^* \rightarrow \mathbb{K}$ est défini par

$$\langle u, v \rangle := \delta_{u,v}, \quad (2.2.5)$$

pour tous $u, v \in L^*$. D'après (2.1.40), le produit \cdot de \mathcal{M}^* , dual du coproduit Δ de \mathcal{M} , admet pour tous $u, v \in L^*$ l'expression suivante étendue par linéarité :

$$u \cdot v = \sum_{w \in L^*} \langle \Delta(w), u \otimes v \rangle w. \quad (2.2.6)$$

Ainsi, un mot w apparaît dans le produit $u \cdot v$ si et seulement si le tenseur $u \otimes v$ apparaît dans $\Delta(w)$. L'apparition de $u \otimes v$ dans $\Delta(w)$ implique $w = u \cdot v$ par définition de Δ . Le produit \cdot correspond donc à la concaténation. De même, d'après (2.1.39), le coproduit Δ^* de \mathcal{M}^* , dual du produit \sqcup de \mathcal{M} , admet pour tout $w \in L^*$ l'expression suivante étendue par linéarité :

$$\Delta^*(w) = \sum_{u,v \in L^*} \langle u \sqcup v, w \rangle u \otimes v. \quad (2.2.7)$$

Ainsi, un tenseur $u \otimes v$ apparaît dans $\Delta^*(w)$ si et seulement si w apparaît dans le produit de mélange de u et de v . De plus, son coefficient est le coefficient avec lequel w apparaît dans $u \sqcup v$. Ceci implique que le coefficient du tenseur $u \otimes v$ dans $\Delta^*(w)$ est le nombre de couples de suites d'entiers $(i_1 < \dots < i_{|u|}, j_1 < \dots < j_{|v|})$ tels que $w_{i_1} = u_1, \dots, w_{i_{|u|}} = u_{|u|}$ et $w_{j_1} = v_1, \dots, w_{j_{|v|}} = v_{|v|}$.

Voici un exemple avec $L := \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$:

$$\begin{aligned} \Delta^*(\mathbf{bacc}) &= 1 \otimes \mathbf{bacc} + \mathbf{b} \otimes \mathbf{acc} + \mathbf{a} \otimes \mathbf{bcc} + 2 \mathbf{c} \otimes \mathbf{bac} + \mathbf{ba} \otimes \mathbf{cc} + 2 \mathbf{bc} \otimes \mathbf{ac} \\ &\quad + 2 \mathbf{ac} \otimes \mathbf{bc} + \mathbf{ba} \otimes \mathbf{cc} + 2 \mathbf{bac} \otimes \mathbf{c} + \mathbf{bcc} \otimes \mathbf{a} + \mathbf{acc} \otimes \mathbf{b} + \mathbf{bacc} \otimes 1. \end{aligned} \quad (2.2.8)$$

Une autre manière de voir ce coproduit se base sur le fait que les mots sur L de taille 1 sont les générateurs algébriques de \mathcal{M}^* . Tout mot peut en effet être obtenu par concaténation de lettres, et aucune relation n'a lieu entre ces générateurs. Maintenant, étant donné que d'après la définition 2.1.5, Δ^* doit être un morphisme d'algèbre, il suffit de définir Δ^* sur les générateurs algébriques de \mathcal{M}^* pour le définir entièrement, et la seule possibilité est

$$\Delta^*(\mathbf{a}) = 1 \otimes \mathbf{a} + \mathbf{a} \otimes 1, \quad (2.2.9)$$

pour toute lettre $\mathbf{a} \in L$.

Une réalisation polynomiale

Rappelons maintenant la réalisation polynomiale de \mathcal{M}^* décrite en [Hiv03]. Celle-ci est extrêmement simple puisque \mathcal{M}^* peut être considérée comme la réalisation polynomiale d'elle-même. En effet, pour tout $\mathbf{a} \in L$, on pose

$$r_L(\mathbf{a}) := \mathbf{a}. \quad (2.2.10)$$

Comme les lettres \mathbf{a} de L sont les générateurs algébriques de \mathcal{M}^* , l'application $r_A : \mathcal{M}^* \rightarrow \mathbb{K}\langle L \rangle$, qui se doit d'être un morphisme d'algèbre, est entièrement définie par (2.2.10). Cette application, clairement injective, est donc une réalisation polynomiale de \mathcal{M}^* en tant qu'algèbre. Maintenant, pour définir $r_{L \oplus L}$, il suffit de décrire les images des éléments $\mathbf{a} \in L$. Il existe pour ce faire un unique choix, et pour tout $\mathbf{a} \in L$, on pose

$$r_{L \oplus L}(\mathbf{a}) := \mathbf{a}' + \mathbf{a}'', \quad (2.2.11)$$

où \mathbf{a}' et \mathbf{a}'' sont les lettres qui correspondent à \mathbf{a} dans la somme ordinale $L \uplus L$. Ainsi, en identifiant, d'après (2.1.52), l'élément $\mathbf{a}' + \mathbf{a}'' \in \mathbb{K}\langle L \oplus L \rangle$ à l'élément $1 \otimes \mathbf{a} + \mathbf{a} \otimes 1 \in \mathbb{K}\langle L \rangle \otimes \mathbb{K}\langle L \rangle$, on a

$$(r_L \otimes r_L) \Delta^*(\mathbf{a}) = r_{L \oplus L}(\mathbf{a}), \quad (2.2.12)$$

ce qui montre que (2.2.11) est bien une réalisation polynomiale de \mathcal{M}^* en tant que cogèbre, impliquant finalement que r_L est une réalisation polynomiale de l'algèbre de Hopf combinatoire \mathcal{M}^* .

Calculons par exemple, pour $L := \{\mathbf{a}, \mathbf{b}\}$, le coproduit de \mathbf{aab} . Il suffit pour cela de réaliser l'élément \mathbf{aab} sur l'alphabet $L \oplus L$. Nous avons

$$\begin{aligned} r_{L \oplus L}(\mathbf{aab}) &= (\mathbf{a} + \mathbf{a}') \cdot (\mathbf{a} + \mathbf{a}') \cdot (\mathbf{b} + \mathbf{b}') \\ &= \mathbf{aab} + \mathbf{aab}' + \mathbf{aa'b} + \mathbf{aa'b}' + \mathbf{a'ab} + \mathbf{a'ab}' + \mathbf{a'a'b} + \mathbf{a'a'b}' \\ &= \mathbf{aab} \otimes 1 + \mathbf{aa} \otimes \mathbf{b}' + \mathbf{ab} \otimes \mathbf{a}' + \mathbf{a} \otimes \mathbf{a'b}' + \mathbf{ab} \otimes \mathbf{a}' + \mathbf{a} \otimes \mathbf{a'b}' \\ &\quad + \mathbf{b} \otimes \mathbf{a'a}' + 1 \otimes \mathbf{a'a'b}' \\ &= 1 \otimes \mathbf{aab} + 2 \mathbf{a} \otimes \mathbf{ab} + \mathbf{aa} \otimes \mathbf{b} + 2 \mathbf{ab} \otimes \mathbf{a} + \mathbf{b} \otimes \mathbf{aa} + \mathbf{aab} \otimes 1, \end{aligned} \quad (2.2.13)$$

qui est égal, comme un simple calcul le montre, à $(r_L \otimes r_L) \Delta^*(\mathbf{aab})$.

Pour plus de détails sur les algèbres de Hopf \mathcal{M} et \mathcal{M}^* , on pourra consulter [Reu93].

2.2.2 L'algèbre de Hopf de Connes-Kreimer

L'algèbre des forêts d'arbres enracinés

Nous considérons ici la classe combinatoire des forêts non ordonnées d'arbres enracinés, que nous appellerons *forêts* dans ce paragraphe par souci de concision. La *taille* d'une forêt est la somme du nombre de nœuds de chacun des arbres qui la constituent. La suite **A000081** de [Slo] commençant par

$$1, 1, 2, 4, 9, 20, 48, 115, 286, 719, 1842, 4766, \quad (2.2.14)$$

énumère les forêts selon leur taille. Étant donné qu'il existe exactement une forêt de taille 0, l'espace vectoriel combinatoire engendré par la classe combinatoire des forêts, que nous notons CK, est bien défini. Les éléments de la *base élémentaire* de CK sont notés \mathbf{E}_F où F est une forêt.

Sur l'espace CK, nous définissons linéairement, pour toutes forêts F_0 et F_1 , le produit \cdot par

$$\mathbf{E}_{F_0} \cdot \mathbf{E}_{F_1} = \mathbf{E}_{F_0 \uplus F_1}, \quad (2.2.15)$$

où $F_0 \uplus F_1$ désigne la forêt constituée des arbres de F_0 et de F_1 . Nous avons ainsi par exemple,

$$\mathbf{E}_{\begin{array}{c} \bullet \\ \diagup \diagdown \\ \bullet \bullet \end{array}} \cdot \mathbf{E}_{\begin{array}{c} \bullet \\ \diagup \diagdown \\ \bullet \bullet \end{array}} = \mathbf{E}_{\begin{array}{c} \bullet \bullet \\ \diagup \diagdown \diagup \diagdown \\ \bullet \bullet \bullet \bullet \end{array}}. \quad (2.2.16)$$

Notons que le produit \cdot est commutatif. En notant \perp la forêt de taille 0, l'élément \mathbf{E}_{\perp} est l'élément neutre de ce produit. La structure (\mathbf{CK}, \cdot) est une algèbre combinatoire. La base des \mathbf{E} est en outre, d'après la définition de \cdot , une base multiplicative. De plus, la famille des éléments de la forme \mathbf{E}_T , où T est une forêt constituée d'un unique arbre, forme une base algébrique de \mathbf{CK} puisqu'il n'existe pas de relations non triviales entre les \mathbf{E}_T — outre celles imposées par le fait que le produit est commutatif — et que tout élément \mathbf{E}_F s'obtient de manière évidente par un produit

$$E_F = E_{T_1} \cdot \dots \cdot E_{T_\ell}, \quad (2.2.17)$$

où $F = T_1 \uplus \dots \uplus T_\ell$, et les T_i sont les arbres qui constituent F . De plus, les éléments indécomposables de **CK** sont les éléments de la forme E_T où T est un arbre.

La cogèbre des forêts d'arbres enracinés

Nous pouvons munir l'espace \mathbf{CK} d'une structure de cogèbre combinatoire de la manière suivante. Soit T un arbre. Un ensemble E de nœuds de T forme une *coupe admissible* de T si

$$x \in E \quad \text{et} \quad y \text{ est un ancêtre de } x \quad \text{impliquent} \quad y \in E. \quad (2.2.18)$$

Nous notons $\text{Adm}(T)$ l'ensemble des coupes admissibles de T . Une coupe admissible E permet de construire à partir de T un couple $T|_E := (T', F)$ où T' est l'arbre T restreint aux nœuds de E et à leurs arêtes adjacentes, et F est la forêt composée des nœuds de T qui ne sont pas dans E ainsi que de leurs arêtes adjacentes. Chaque composante connexe de F forme un arbre dont la racine est son unique nœud qui connectait dans T la racine de T par le chemin le plus court. La figure 2.1 montre une coupe admissible et le couple que l'on obtient à partir de celle-ci.

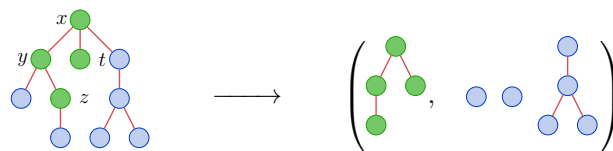


FIGURE 2.1 – Un arbre enraciné T , et le couple $T|_E := (T', F)$ où $E := \{x, y, z, t\}$ est une coupe admissible de Connes-Kreimer.

Soit le coproduit défini linéairement pour tout arbre T par

$$\Delta(\mathbf{E}_T) := \sum_{\substack{E \in \text{Adm}(T) \\ (T', F) := T|_E}} \mathbf{E}_{T'} \otimes \mathbf{E}_F. \quad (2.2.19)$$

L'expression (2.2.19), définie sur les générateurs algébriques de \mathbf{CK} , définit en réalité un coproduit Δ sur \mathbf{CK} tout entier car Δ se doit d'être un morphisme d'algèbre. En effet, si F est une forêt telle que

$$\mathbf{E}_F = \mathbf{E}_{T_1} \cdot \dots \cdot \mathbf{E}_{T_\ell}, \quad (2.2.20)$$

où les T_i sont des arbres, on calcule $\Delta(E_F)$ par

$$\Delta(E_F) = \Delta(E_{T_1}) \odot \cdots \odot \Delta(E_{T_\ell}), \quad (2.2.21)$$

où \odot désigne l'application $I \otimes \omega \otimes I$ définie en conformité avec (2.1.30). Voici un exemple de

coproduit, où l'on note l'élément neutre E_\perp par 1 :

$$\Delta \left(\begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and three children} \end{array} \right) = 1 \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and three children} \end{array} + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and two children} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and two children} \end{array} + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \\ + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} + \begin{array}{c} \text{E} \\ \text{diagram of a tree with root E and one child} \end{array} \otimes 1. \quad (2.2.22)$$

Notons que cet exemple montre que Δ n'est pas cocommutatif. La structure (CK, Δ) est une cogèbre combinatoire.

L'algèbre de Hopf des forêts d'arbres enracinés

Le produit \cdot et le coproduit Δ définis sur CK forment, comme l'ont montré Connes et Kreimer [CK98], une algèbre de Hopf combinatoire. Cette dernière est connue sous le nom d'*algèbre de Hopf de Connes-Kreimer*. Donnons ici, afin d'être exhaustif, l'antipode S de CK , mais sans explications supplémentaires — pour plus de détails, voir par exemple [Foi02]. Il est défini linéairement pour tout arbre T par

$$S(E_T) = -E_T - \sum_{\substack{E \in \text{Adm}(T) \\ (T', F) := T|_E}} (-1_{\mathbb{K}})^{|T|} E_{T' \uplus F}. \quad (2.2.23)$$

Comme tout antipode est un antihomomorphisme d'algèbre, de la même manière que pour le coproduit, l'expression de S donnée en (2.2.23) permet de calculer l'antipode de tout élément de CK . Remarquons de plus que d'après la définition du coproduit Δ , la base des E de CK est du côté gauche.

Pour terminer avec cet exemple, notons que l'algèbre de Hopf de Connes-Kreimer admet un certain nombre de variantes — qui sont plus précisément des généralisations. Il existe par exemple une version non commutative de CK [Foi09], [FNT10] dont les bases sont indexées par des forêts ordonnées d'arbres plans, et une version décorée de CK [Foi02], [Foi09] dont les bases sont indexées par des forêts d'arbres étiquetés sur un alphabet non vide. Une version non commutative de cette dernière, dont les bases sont indexées par des forêts ordonnées d'arbres plans enracinés étiquetés, est également étudiée dans ces références. Une réalisation polynomiale de cette algèbre de Hopf fut découverte récemment par Foissy, Novelli, et Thibon [FNT10] et du même coup, une réalisation polynomiale de CK également.

2.2.3 L'algèbre de Hopf des fonctions quasi-symétriques libres

L'algèbre des permutations

Notons **FQSym** l'espace vectoriel combinatoire engendré par la classe combinatoire des permutations. Les éléments de la *base fondamentale* de **FQSym** sont notés \mathbf{F}_σ où σ est une permutation. Les dimensions composante homogène par composante homogène de cet espace forment la suite **A000142** de [Slo] qui commence par

$$1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880, \quad (2.2.24)$$

et qui dénombre bien entendu les permutations selon leur taille — le n^{e} terme de cette suite est $n!$.

Sur l'espace **FQSym**, on définit linéairement pour toutes permutations σ et ν le produit \cdot par

$$\mathbf{F}_\sigma \cdot \mathbf{F}_\nu := \sum_{\pi \in \sigma \boxtimes \nu} \mathbf{F}_\pi. \quad (2.2.25)$$

Nous avons ainsi par exemple

$$\begin{aligned} \mathbf{F}_{132} \cdot \mathbf{F}_{12} = & \mathbf{F}_{13245} + \mathbf{F}_{13425} + \mathbf{F}_{13452} + \mathbf{F}_{14325} + \mathbf{F}_{14352} \\ & + \mathbf{F}_{14532} + \mathbf{F}_{41325} + \mathbf{F}_{41352} + \mathbf{F}_{41532} + \mathbf{F}_{45132}. \end{aligned} \quad (2.2.26)$$

La structure (\mathbf{FQSym}, \cdot) est une algèbre combinatoire.

La cogèbre des permutations

L'espace \mathbf{FQSym} peut également être muni d'un coproduit Δ , défini linéairement pour toute permutation π par

$$\Delta(\mathbf{F}_\pi) := \sum_{\pi=u \cdot v} \mathbf{F}_{\text{std}(u)} \otimes \mathbf{F}_{\text{std}(v)}. \quad (2.2.27)$$

Nous avons par exemple

$$\begin{aligned} \Delta(\mathbf{F}_{24153}) = & 1 \otimes \mathbf{F}_{24153} + \mathbf{F}_1 \otimes \mathbf{F}_{3142} + \mathbf{F}_{12} \otimes \mathbf{F}_{132} \\ & + \mathbf{F}_{231} \otimes \mathbf{F}_{21} + \mathbf{F}_{2314} \otimes \mathbf{F}_1 + \mathbf{F}_{24153} \otimes 1. \end{aligned} \quad (2.2.28)$$

La structure (\mathbf{FQSym}, Δ) est une cogèbre combinatoire.

L'algèbre de Hopf des permutations

Le produit \cdot et le coproduit Δ définis sur \mathbf{FQSym} forment, comme l'on montré Malvenuto et Reutenauer [MR95], une bigèbre combinatoire et donc une algèbre de Hopf combinatoire. Pour être exact, dans leur construction initiale, Malvenuto et Reutenauer ont construit le dual \mathbf{FQSym}^* de \mathbf{FQSym} . Une formule explicite pour l'antipode que nous ne détaillerons pas est établie en [AS05].

L'algèbre de Hopf duale et autodualité

Voici la description de \mathbf{FQSym}^* . Pour toutes permutations σ et ν , le produit \cdot de \mathbf{FQSym}^* vérifie

$$\mathbf{F}_\sigma^* \cdot \mathbf{F}_\nu^* = \sum_{\substack{u \cdot v \in \mathfrak{S} \\ \text{std}(u) = \sigma \\ \text{std}(v) = \nu}} \mathbf{F}_\pi^*. \quad (2.2.29)$$

Le coproduit Δ de \mathbf{FQSym}^* vérifie quant à lui, pour toute permutation π ,

$$\Delta(\mathbf{F}_\pi^*) = \sum_{\substack{0 \leq i \leq n \\ \sigma := \pi|_{[1, i]} \\ \nu := \text{std}(\pi|_{[i+1, n]}}} \mathbf{F}_\sigma^* \otimes \mathbf{F}_\nu^*. \quad (2.2.30)$$

L'algèbre de Hopf \mathbf{FQSym} est autoduale. En effet, en posant $\psi : \mathbf{FQSym} \rightarrow \mathbf{FQSym}^*$, le morphisme de Hopf défini linéairement pour toute permutation σ par

$$\psi(\mathbf{F}_\sigma) := \mathbf{F}_{\sigma^{-1}}^*, \quad (2.2.31)$$

est un isomorphisme.

Autres bases, liberté et générateurs algébriques

Il existe plusieurs bases intéressantes de **FQSym**, construites en imitant les définitions de diverses bases de l'algèbre des fonctions symétriques [Mac95], dont le comportement vis à vis du produit dépend de propriétés combinatoires du permuttoèdre [DHNT08]. La *base élémentaire* de **FQSym** est définie pour toute permutation σ par

$$\mathbf{E}^\sigma := \sum_{\sigma \leq_p \sigma'} \mathbf{F}_{\sigma'}. \quad (2.2.32)$$

Similairement, la *base homogène* de **FQSym** est définie par

$$\mathbf{H}^\sigma := \sum_{\sigma' \leq_p \sigma} \mathbf{F}_{\sigma'}. \quad (2.2.33)$$

Ces deux bases sont multiplicatives puisque le produit vérifie, pour toutes permutations σ et ν ,

$$\mathbf{E}^\sigma \cdot \mathbf{E}^\nu = \mathbf{E}^{\sigma \setminus \nu}, \quad (2.2.34)$$

et

$$\mathbf{H}^\sigma \cdot \mathbf{H}^\nu = \mathbf{H}^{\sigma \setminus \nu}. \quad (2.2.35)$$

Ces bases permettent de montrer que **FQSym** est libre et que les éléments \mathbf{E}^σ (resp. \mathbf{H}^σ) où σ est une permutation connexe (resp. anti-connexe) forment une famille de générateurs algébriques. Ainsi, les dimensions des générateurs algébriques de **FQSym** constituent la suite **A003319** de [Slo] qui commence par

$$0, 1, 1, 3, 13, 71, 461, 3447, 29093, 273343, \quad (2.2.36)$$

et qui dénombre les permutations connexes (ou les permutations anti-connexes).

Une autre base usuelle de **FQSym** est la base des **G** définie pour toute permutation σ par $\mathbf{G}_\sigma := \mathbf{F}_{\sigma^{-1}}$. D'après l'expression de l'isomorphisme (2.2.31) entre **FQSym** et son dual, le produit et le coproduit dans la base des **G** vérifient respectivement (2.2.29) et (2.2.30) en substituant dans ces deux expressions les symboles \mathbf{F}^* par **G**.

Une réalisation polynomiale

Rappelons que $A = \{\mathbf{a}_i : i \geq 1\}$ est un alphabet totalement ordonné par la relation \leq où $\mathbf{a}_i \leq \mathbf{a}_j$ pour tous $1 \leq i \leq j$. Soit l'application linéaire $r_A : \mathbf{FQSym} \rightarrow \mathbb{K}\langle A \rangle$ définie dans la base des **G** pour toute permutation σ par

$$r_A(\mathbf{G}_\sigma) := \sum_{\substack{u \in A^* \\ \text{std}(u) = \sigma}} u. \quad (2.2.37)$$

Hivert, Duchamp et Thibon ont montré [DHT02] (voir aussi [Hiv03]) que r_A est une réalisation polynomiale de l'algèbre **FQSym**. L'image de **FQSym** par r_A forme l'*algèbre des fonctions quasi-symétriques libres*. Nous avons ainsi par exemple

$$r_A(\mathbf{G}_1) = \sum_{\mathbf{a} \in A} \mathbf{a} = \mathbf{a}_1 + \mathbf{a}_2 + \mathbf{a}_3 + \cdots, \quad (2.2.38)$$

$$r_A(\mathbf{G}_{4213}) = \sum_{1 \leq i < j \leq k < \ell} \mathbf{a}_\ell \mathbf{a}_j \mathbf{a}_i \mathbf{a}_k = \mathbf{a}_3 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_4 \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_3 + \cdots. \quad (2.2.39)$$

En outre, l'application r_A est également une réalisation de la cogèbre **FQSym**. En effet, si $B := \{b_i : i \geq 1\}$ est un alphabet totalement ordonné par $b_i \leq b_j$ pour tous $1 \leq i \leq j$, nous avons par exemple

$$\begin{aligned}
 r_{A \oplus B}(\mathbf{G}_{31524}) = & \sum_{i \leq j < k \leq \ell < r} a_k a_i a_r a_j a_\ell + \sum_{\substack{i \leq j < k \leq \ell \\ r}} a_k a_i b_r a_j a_\ell + \sum_{\substack{i \leq j < k \\ \ell < r}} a_k a_i b_r a_j b_\ell \\
 & + \sum_{\substack{i \leq j \\ k \leq \ell < r}} b_k a_i b_r a_j b_\ell + \sum_{\substack{i \\ j < k \leq \ell < r}} b_k a_i b_r b_j b_\ell + \sum_{i \leq j < k \leq \ell < r} b_k b_i b_r b_j b_\ell.
 \end{aligned} \tag{2.2.40}$$

On s'aperçoit que si l'on applique $r_A \otimes r_B$ à l'expression (2.2.28), on retrouve bien (2.2.40) puisque les lettres de A et de B commutent entre elles.

Terminons ce paragraphe en citant [Mal93], [DHT02], [AS05] et [DHNT08] qui constituent les principales références à propos des travaux réalisés sur **FQSym**.

Chapitre 3

Opérades

Les opérades furent introduites dans les années 1970 à peu de choses près au même moment par May [May72] et par Boardman et Vogt [BV73] dans le contexte de la topologie algébrique et l'étude des espaces de lacets. Cette nouvelle notion tombait progressivement dans l'oubli lorsque, à partir des années 1990, elle trouva des applications dans d'autres domaines [Lod95], [Sta99b], [MSS02], notamment en algèbre, en physique et en combinatoire. Quelques références à la fois générales et introductives sur les opérades sont [GK94], [Mar06], ainsi que [LV10].

De manière simplifiée, une opérade est une structure algébrique qui contient des opérateurs abstraits pouvant se composer pour en former de plus gros. Par exemple, composer un opérateur x d'arité n avec un autre opérateur y d'arité m donne un opérateur d'arité $n + m - 1$ car l'opérateur obtenu dispose des entrées de y ainsi que celles de x excepté de l'une d'entre elles — celle utilisée pour réaliser la composition. Une algèbre sur une opérade est un espace vectoriel sur lequel ces opérateurs agissent, c'est-à-dire qu'ils permettent de calculer un élément en sortie sur l'entrée de plusieurs éléments de l'espace vectoriel. Le point fondamental est que les relations qui existent entre les opérateurs de l'opérade impliquent des relations entre les éléments de l'algèbre. Par exemple, toute algèbre sur l'opérade associative (voir [AL07]) possède un opérateur, un produit, qui est associatif. De même, toute algèbre sur l'opérade de Lie possède un crochet de Lie qui vérifie la relation de Jacobi et est antisymétrique. Ainsi, chaque type d'algèbre est gouverné par une opérade. L'un des points forts de cette théorie est qu'il devient alors possible de comparer différents types d'algèbres en passant par les opérades, ce qui se fait par l'intermédiaire de morphismes d'opérade. Citons à ce propos [Zin10] qui répertorie une vaste gamme de types d'algèbres ainsi que leurs opérades correspondantes.

Notre traitement des opérades sera quelque peu différent puisque nous avons une vision purement combinatoire de ces structures. En suivant notre même approche dans l'étude des algèbres de Hopf qui consiste à les regarder comme encodant l'*assemblage* et le *désassemblage* d'objets combinatoires avec une notion de compatibilité, les opérades sont vues comme des structures permettant d'encoder le *greffage* d'objets combinatoires. La différence avec la notion d'assemblage propre aux algèbres de Hopf est que l'on peut choisir un endroit dans l'objet pour réaliser ladite greffe. Cette vision met en évidence le fait qu'un objet combinatoire est constitué de *secteurs de substitution* — jouant le rôle d'entrées d'opérateurs — sur lesquels d'autres objets peuvent y être greffés. Cette opération est incarnée par les opérateurs de substitution partielle des opérades.

Dans ces dernières années, un grand nombre d'opérades ont été définies et étudiées, et force est de constater que la plupart sont définies sur des objets combinatoires — ou des espaces vectoriels engendrés par une classe combinatoire. Une large palette d'opérades mettant en jeu différentes espèces d'arbres occupent une place de choix en combinatoire algébrique [Cha08].

Citons par exemple l'opérade dendriforme [Lod01] de Loday et l'opérade dupliciale [BF03] de Brouder et Frabetti toutes deux sur les arbres binaires. Ou encore, l'opérade pré-Lie [CL01] de Chapoton et Livernet et l'opérade non associative permutative [Liv06] de Livernet toutes deux définies sur les arbres enracinés. Citons également des opérades sur des objets combinatoires plus exotiques comme l'opérade des plantes et des arbres non croisés [Cha06b], des forêts d'arbres binaires dont les feuilles sont étiquetées [Cha04], et des arbustes [Cha10], toutes construites par Chapoton. Un point remarquable dans ces exemples est que la substitution partielle s'exprime par des algorithmes combinatoires et met en évidence certaines propriétés des objets mis en jeu.

Dans ce chapitre, nous posons les concepts de base sur les opérades que nous utiliserons dans les chapitres 6 et 7. Nous définissons ainsi dans le paragraphe 3.1 les opérades dans la catégorie des espaces vectoriels, les notions de morphismes d'opérade, d'idéaux et de quotient. Nous donnons également une description de l'opérade libre sur un ensemble de générateurs et de la présentation d'une opérade comme un quotient de cette dernière. De plus, nous rappelons ce qu'est une algèbre sur une opérade. Nous donnons dans le paragraphe 3.2, quatre exemples d'opérades qui nous paraissent fondamentales : l'opérade commutative et associative, l'opérade associative, l'opérade de Lie et l'opérade dendriforme. Nous terminons par le paragraphe 3.3 en rappelant une construction classique qui, à une opérade ensembliste, associe un groupe, puis deux algèbres de Hopf, l'une commutative et l'autre non.

3.1 Définitions et propriétés de base

3.1.1 Définitions

Substitutions dans une permutation

Nous avons avant tout besoin de décrire une opération sur les permutations. Soient $\sigma \in \mathfrak{S}_n$ et $\nu \in \mathfrak{S}_m$ deux permutations. Pour tout $i \in [n]$, la *substitution* de ν en position i dans σ , notée $B_i(\sigma, \nu)$, est la permutation π qui vérifie

$$\pi := \pi_1 \cdot \dots \cdot \pi_{i-1} \cdot (\sigma_i + \nu_1 - 1) \cdot \dots \cdot (\sigma_i + \nu_m - 1) \cdot \pi_{i+1} \cdot \dots \cdot \pi_n, \quad (3.1.1)$$

où

$$\pi_j := \begin{cases} \sigma_j & \text{si } \sigma_j < \sigma_i, \\ \sigma_j + m - 1 & \text{sinon.} \end{cases} \quad (3.1.2)$$

Nous avons ainsi par exemple $B_3(5341672, 231) = 735641892$.

Opérades

Définition 3.1.1. Une opérade est un espace vectoriel gradué \mathcal{P} de la forme

$$\mathcal{P} := \bigoplus_{n \geq 1} \mathcal{P}(n), \quad (3.1.3)$$

munie pour tout $i \geq 1$ d'applications linéaires \circ_i , nommées opérateurs de substitution partielle, qui sont pour tous $n \geq i$ et $m \geq 1$ de la forme

$$\circ_i : \mathcal{P}(n) \otimes \mathcal{P}(m) \rightarrow \mathcal{P}(n + m - 1), \quad (3.1.4)$$

et qui vérifient pour tous $x \in \mathcal{P}(n)$, $y \in \mathcal{P}(m)$, $z \in \mathcal{P}(k)$, $i \in [n]$ et $j \in [m]$ la relation d'associativité en série :

$$(x \circ_i y) \circ_{i+j-1} z = x \circ_i (y \circ_j z), \quad (3.1.5)$$

et pour tous $1 \leq i < j \leq n$, la relation d'associativité en parallèle :

$$(x \circ_i y) \circ_{j+m-1} z = (x \circ_j z) \circ_i y. \quad (3.1.6)$$

Il existe de plus un élément $\mathbf{1} \in \mathcal{P}(1)$, appelé unité, tel que pour tout $x \in \mathcal{P}(n)$ et $i \in [n]$,

$$\mathbf{1} \circ_1 x = x \circ_i \mathbf{1} = x. \quad (3.1.7)$$

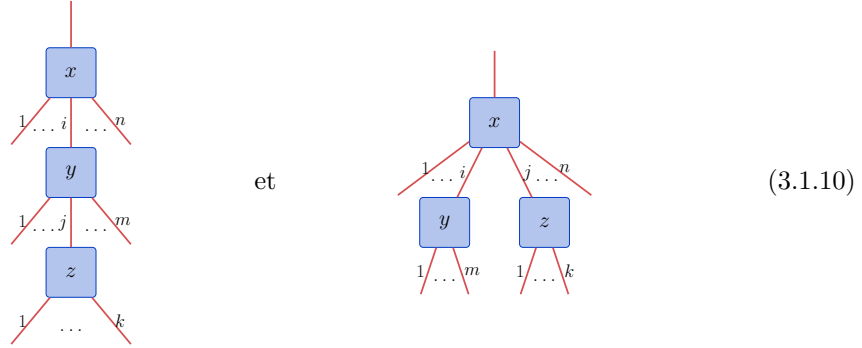
Chaque espace vectoriel $\mathcal{P}(n)$ est muni d'une action du groupe symétrique linéaire à gauche

$$\cdot : \mathcal{P}(n) \times \mathfrak{S}_n \rightarrow \mathcal{P}(n), \quad (3.1.8)$$

qui vérifie, pour tous $x \in \mathcal{P}(n)$, $\sigma \in \mathfrak{S}_n$, $y \in \mathcal{P}(m)$, $\nu \in \mathfrak{S}_m$ et $i \in [n]$ la relation

$$(x \cdot \sigma) \circ_i (y \cdot \nu) = (x \circ_{\sigma_i} y) \cdot B_i(\sigma, \nu). \quad (3.1.9)$$

Les éléments de $\mathcal{P}(n)$ peuvent être vus comme des opérateurs qui disposent de n entrées et d'une sortie. De ce fait, si x est un élément de $\mathcal{P}(n)$, nous dirons que l'arité de x est n . Les relations imposées dans la définition 3.1.1 peuvent se comprendre en termes d'assemblages d'opérateurs. En effet, en ce qui concerne les relations d'associativité en série et en parallèle, il suffit de voir que l'on peut assembler trois opérateurs x , y et z d'exactly deux manières différentes, au renommage près des opérateurs :



Pour comprendre la relation d'associativité en série, on peut observer qu'il existe deux manières, à partir des éléments x , y , et z d'obtenir l'opérateur de gauche de (3.1.10) : on peut en effet commencer par greffer y à x , et ensuite greffer z à l'opérateur obtenu, ou bien, on peut greffer z à y , et ensuite greffer l'opérateur obtenu à x . Le fait que ces substitutions partielles aboutissent à un même élément est encodé par (3.1.5).

De la même manière, pour comprendre la relation d'associativité en parallèle, on peut observer que l'on a deux manières, à partir des éléments x , y , et z d'obtenir l'opérateur de droite de (3.1.10) : on peut en effet commencer par greffer y à x , et ensuite greffer z à l'opérateur obtenu, ou bien, on peut greffer z à x , et ensuite greffer y à l'opérateur obtenu. Ici, le fait que ces substitutions partielles aboutissent à un même élément est encodé par (3.1.6).

En outre, la relation (3.1.9) spécifiant la compatibilité entre les opérateurs de substitution partielle et l'action du groupe symétrique peut se comprendre de la même façon, en sachant que l'action du groupe symétrique sur les éléments de l'opérade se traduit par une permutation des entrées des opérateurs qui les représentent.

Opérateurs de substitution complète

On peut associer à toute opérade \mathcal{P} un *opérateur de substitution complète* \circ de la forme

$$\circ : \mathcal{P}(n) \otimes \mathcal{P}(m_1) \otimes \cdots \otimes \mathcal{P}(m_n) \rightarrow \mathcal{P}(m_1 + \cdots + m_n), \quad (3.1.11)$$

$$(x, y_1, \dots, y_n) \mapsto x \circ [y_1, \dots, y_n], \quad (3.1.12)$$

défini pour tous $n, m_1, \dots, m_n \geq 1$ à partir des opérateurs de substitution partielle par

$$x \circ [y_1, \dots, y_n] := (\dots ((x \circ_n y_n) \circ_{n-1} y_{n-1}) \dots) \circ_1 y_1, \quad (3.1.13)$$

où $x \in \mathcal{P}(n)$, $y_1 \in \mathcal{P}(m_1)$, \dots , $y_n \in \mathcal{P}(m_n)$. Les relations d'associativité en série et en parallèle données en (3.1.5) et (3.1.6) se résument alors par l'unique relation

$$(x \circ [y_1, \dots, y_n]) \circ [z_{1,1}, \dots, z_{1,m_1}, \dots, z_{n,1}, \dots, z_{n,m_n}] = x \circ [y_1 \circ [z_{1,1}, \dots, z_{1,m_1}], \dots, y_n \circ [z_{n,1}, \dots, z_{n,m_n}]], \quad (3.1.14)$$

pour tous $x \in \mathcal{P}(n)$, $y_1 \in \mathcal{P}(m_1)$, \dots , $y_n \in \mathcal{P}(m_n)$, et $z_{1,1}, \dots, z_{1,m_1}, \dots, z_{n,1}, \dots, z_{n,m_n} \in \mathcal{P}$. La relation (3.1.7) se traduit quant à elle en la relation

$$\mathbf{1} \circ [x] = x \circ \underbrace{[\mathbf{1}, \dots, \mathbf{1}]_n} = x, \quad (3.1.15)$$

pour tout $x \in \mathcal{P}$.

Inversement, à partir d'un opérateur de substitution complète \circ vérifiant (3.1.14) et d'une unité $\mathbf{1}$ vérifiant (3.1.15), il est possible de retrouver les opérateurs \circ_i de substitution partielle en posant

$$x \circ_i y := x \circ \underbrace{[\mathbf{1}, \dots, \mathbf{1}]_{i-1}}_{i-1}, y, \underbrace{[\mathbf{1}, \dots, \mathbf{1}]_{n-i}}_{n-i}, \quad (3.1.16)$$

où $x \in \mathcal{P}(n)$, $y \in \mathcal{P}(m)$ et $i \in [n]$.

Il y a de ce fait deux manières équivalentes de voir une même opérade : l'une par l'intermédiaire d'un opérateur de substitution complète et l'autre par le biais d'opérateurs de substitution partielle.

Sous-opérades, morphismes, quotients et séries de Hilbert

Soit $(\mathcal{P}, \circ^{\mathcal{P}}, \mathbf{1}^{\mathcal{P}}, \cdot^{\mathcal{P}})$ une opérade. Un espace vectoriel \mathcal{P}' est une *sous-opérade* de \mathcal{P} si pour tout $n \geq 1$, $\mathcal{P}'(n)$ est un sous-espace vectoriel de $\mathcal{P}(n)$, $\mathbf{1}^{\mathcal{P}} \in \mathcal{P}'(1)$ et pour tout $x \in \mathcal{P}'(n)$, $y \in \mathcal{P}'(m)$, $i \in [n]$ et $\sigma \in \mathfrak{S}_n$, on a $x \circ_i y \in \mathcal{P}'(n+m-1)$ et $x \cdot^{\mathcal{P}} \sigma \in \mathcal{P}'(n)$. La sous-opérade de \mathcal{P} engendrée par un ensemble E d'éléments de \mathcal{P} , notée $\langle E \rangle_{\mathcal{P}}$ — ou simplement $\langle E \rangle$ s'il n'y a pas d'ambiguïté — est la plus petite sous-opérade de \mathcal{P} qui contient E . Nous dirons que \mathcal{P} est *finiment engendrée* s'il existe un ensemble fini E tel que $\mathcal{P} = \langle E \rangle$.

Soit maintenant $(\mathcal{Q}, \circ^{\mathcal{Q}}, \mathbf{1}^{\mathcal{Q}}, \cdot^{\mathcal{Q}})$ une autre opérade. Une application $\phi : \mathcal{P} \rightarrow \mathcal{Q}$ est un *morphisme d'opérade* si $\phi(x) \in \mathcal{Q}(n)$, $\phi(\mathbf{1}^{\mathcal{P}}) = \mathbf{1}^{\mathcal{Q}}$,

$$\phi(x \circ_i^{\mathcal{P}} y) = \phi(x) \circ_i^{\mathcal{Q}} \phi(y), \quad (3.1.17)$$

et

$$\phi(x \cdot^{\mathcal{P}} \sigma) = \phi(x) \cdot^{\mathcal{Q}} \sigma, \quad (3.1.18)$$

pour tous $x \in \mathcal{P}(n)$, $y \in \mathcal{P}(m)$, $i \in [n]$ et $\sigma \in \mathfrak{S}_n$.

Un sous-espace vectoriel V de \mathcal{P} est un *idéal* de \mathcal{P} si

$$x \circ_i^{\mathcal{P}} y \in V, \quad (3.1.19)$$

pour tous $x \in \mathcal{P}(n)$, $y \in \mathcal{P}(m)$ et $i \in [n]$, lorsque x ou y est un élément de V et

$$x \cdot^{\mathcal{P}} \sigma \in V, \quad (3.1.20)$$

pour tous $x \in \mathcal{P}(n)$ et $\sigma \in \mathfrak{S}_n$, lorsque x est un élément de V . Le *quotient* de \mathcal{P} par l'idéal V est l'opérade \mathcal{P}/V avec la substitution partielle \circ_i définie linéairement par

$$\hat{x} \circ_i \hat{y} := \tau(x \circ_i^{\mathcal{P}} y), \quad (3.1.21)$$

et l'action du groupe symétrique définie linéairement par

$$\hat{x} \cdot \sigma := \tau(x \cdot^{\mathcal{P}} \sigma), \quad (3.1.22)$$

pour tous $\hat{x} \in \mathcal{P}(n)/V$, $\hat{y} \in \mathcal{P}(m)/V$, $i \in [n]$ et $\sigma \in \mathfrak{S}_n$, où $\tau : \mathcal{P} \rightarrow \mathcal{P}/V$ est la projection canonique et x et y sont des éléments de \mathcal{P} tels que $\tau(x) = \hat{x}$ et $\tau(y) = \hat{y}$.

Si les espaces vectoriels $\mathcal{P}(n)$ de l'opérade \mathcal{P} sont de dimensions finies pour tout $n \geq 1$, on associe alors à \mathcal{P} sa *série de Hilbert* $F_{\mathcal{P}}^e(t)$, la série génératrice exponentielle définie par

$$F_{\mathcal{P}}^e(t) := \sum_{n \geq 1} \frac{1}{n!} \dim \mathcal{P}(n) t^n. \quad (3.1.23)$$

Opérades ensemblistes

Lorsqu'il existe une base de \mathcal{P} telle que la substitution partielle de deux éléments de base, l'unité de \mathcal{P} et l'action de toute permutation sur un élément de base s'expriment tous par un unique élément de base ayant $1_{\mathbb{K}}$ comme coefficient, nous dirons que \mathcal{P} est une *opérade ensembliste*. Dans ce cas, nous pourrions considérer que l'opérade est définie sur une suite d'ensembles $E(n)$, $n \geq 1$, dont les éléments de chaque $E(n)$ sont les éléments de base de $\mathcal{P}(n)$. Il nous arrivera ainsi dans la suite de définir des opérades ensemblistes \mathcal{Q} où les $\mathcal{Q}(n)$ ne sont pas des espaces vectoriels mais des ensembles. On retrouve à partir d'une telle opérade \mathcal{Q} une opérade conforme à la définition 3.1.1 en considérant l'espace vectoriel libre sur l'ensemble \mathcal{Q} .

Opérades non symétriques

Une opérade \mathcal{P} est *non symétrique* — ou ns en abrégé — si elle n'est pas munie d'une action du groupe symétrique. Si les espaces vectoriels $\mathcal{P}(n)$ sont de dimensions finies pour tout $n \geq 1$, on associe alors à \mathcal{P} sa *série de Hilbert* $F_{\mathcal{P}}(t)$, la série génératrice ordinaire définie par

$$F_{\mathcal{P}}(t) := \sum_{n \geq 1} \dim \mathcal{P}(n) t^n. \quad (3.1.24)$$

On obtient à partir d'une opérade sa *version non symétrique* simplement en oubliant l'action du groupe symétrique. Inversement, à partir d'une opérade ns $(\mathcal{P}, \circ^{\mathcal{P}}, \mathbf{1}^{\mathcal{P}})$, on obtient une opérade \mathcal{Q} , la *symétrisée* de \mathcal{P} où

$$\mathcal{Q}(n) := \mathcal{P}(n) \otimes \text{Vect}(\mathfrak{S}_n). \quad (3.1.25)$$

L'opérateur de substitution partielle de \mathcal{Q} est défini linéairement par

$$(x \otimes \sigma) \circ_i (y \otimes \nu) := (x \circ_i^{\mathcal{P}} y) \otimes B_i(\sigma, \nu), \quad (3.1.26)$$

où $x \in \mathcal{P}(n)$, $\sigma \in \mathfrak{S}_n$, $y \in \mathcal{P}(m)$, $\nu \in \mathfrak{S}_m$ et $i \in [n]$. L'unité de \mathcal{Q} est $\mathbf{1} := \mathbf{1}^{\mathcal{P}} \otimes \epsilon$, et l'action du groupe symétrique de \mathcal{Q} est définie linéairement à gauche pour tous $x \in \mathcal{P}(n)$, $\sigma \in \mathfrak{S}_n$ et $\nu \in \mathfrak{S}_n$ par

$$(x \otimes \sigma) \cdot \nu := x \otimes (\sigma \cdot \nu), \quad (3.1.27)$$

où le symbole \cdot dans le membre droit de (3.1.27) désigne la composition usuelle des permutations. Notons que dans le cas général, une opérade \mathcal{P} est différente de la symétrisée de sa version non symétrique.

Remarquons que si \mathcal{P} admet une série de Hilbert $F_{\mathcal{P}}(t)$, alors \mathcal{Q} admet également une série de Hilbert $F_{\mathcal{Q}}^e(t)$. En outre, comme nous avons $\dim \mathcal{Q}(n) = n! \dim \mathcal{P}(n)$ pour tout $n \geq 1$, celle-ci vérifie

$$F_{\mathcal{Q}}^e(t) = F_{\mathcal{P}}(t). \quad (3.1.28)$$

Opérades combinatoires

Dans ce mémoire, nous considérerons plus particulièrement la sous-classe des *opérades combinatoires* :

Définition 3.1.2. Une opérade \mathcal{P} est combinatoire s'il existe un espace vectoriel combinatoire V tel que $\dim V^{(1)} = 1$ et $\mathcal{P}(n) = V^{(n)}$ pour tout $n \geq 1$. Nous dirons que la classe combinatoire sous-jacente à \mathcal{P} est la classe combinatoire sous-jacente à V .

Notons que d'après la définition 1.1.1 du chapitre 1, toute opérade combinatoire \mathcal{P} admet une série de Hilbert $F_{\mathcal{P}}^e(t)$ puisque les espaces vectoriels $\mathcal{P}(n)$ sont de dimensions finies. Celle-ci est de la forme

$$F_{\mathcal{P}}^e(t) = t + \sum_{n \geq 2} \frac{1}{n!} [t^n] \mathcal{S}_C(t), \quad (3.1.29)$$

où C est la classe combinatoire sous-jacente à \mathcal{P} . Notons que d'après la définition 3.1.2, le coefficient de t dans $F_{\mathcal{P}}^e(t)$ est bien 1.

La plupart des opérades combinatoires peuvent s'implanter sans difficulté, par exemple en Sage [S⁺11]. Il suffit en effet de spécifier la classe combinatoire sous-jacente et de programmer l'opérateur de substitution partielle de deux éléments de base ainsi que l'action du groupe symétrique. Le système étend automatiquement la substitution partielle par linéarité. Un bon nombre d'implantations ont été réalisées dans ce travail pour des expérimentations.

3.1.2 Opérades libres, présentations par générateurs et relations

Nous donnons ici une description simplifiée et suffisante dans notre contexte de l'opérade libre. Pour une construction complète et plus générale, on pourra consulter [Mar06] et [LV10].

Opérades libres sur un ensemble

Soit $E := \uplus_{n \geq 1} E(n)$ un ensemble. Les éléments de $E(n)$ peuvent se voir comme des *opérateurs* d'arité n . Nous considérons l'espace vectoriel $\mathcal{F}(E)$ engendré par l'ensemble des arbres plans enracinés dont les nœuds qui ne possèdent pas de fils — les feuilles — ne sont pas étiquetés et les nœuds qui possèdent exactement n fils sont étiquetés sur $E(n)$. Nous appellerons dans ce paragraphe simplement *arbres* ces objets et leur *taille* est leur nombre de feuilles. Le sous-espace vectoriel de $\mathcal{F}(E)$ restreint aux arbres de taille n est noté $\mathcal{F}(E)(n)$. Un arbre peut se voir comme un arbre syntaxique d'une expression où les opérateurs sont les étiquettes de ses nœuds et les variables en entrée correspondent à ses feuilles. Ainsi, par exemple, à l'arbre



$$(3.1.30)$$

correspond l'expression

$$\mathbf{d}(\mathbf{b}(v, w), x, \mathbf{c}(y, z)), \quad (3.1.31)$$

où les nœuds d'étiquettes \mathbf{b} et \mathbf{c} correspondent à des opérateurs d'arité deux, le nœud d'étiquette \mathbf{d} à un opérateur d'arité trois, et v, w, x, y et z sont des variables.

La substitution partielle de deux arbres S et T en position i est l'arbre obtenu en *greffant* la racine de T sur la i^{e} feuille de S en partant de la gauche. Nous notons $S \circ_i T$ l'arbre ainsi obtenu. L'arbre constitué d'une unique feuille est l'unité pour cette substitution partielle. Celui-ci est de taille 1 et est noté $\mathbf{1}$. En étendant la définition de l'opérateur de substitution partielle par linéarité, l'*opérade ns libre* sur l'ensemble E est l'opérade $(\mathcal{F}(E), \circ, \mathbf{1})$. Donnons à présent un exemple. Fixons $E(1) := \{\mathbf{a}\}$, $E(2) := \{\mathbf{b}, \mathbf{c}\}$, $E(3) := \{\mathbf{d}\}$ et $E := E(1) \uplus E(2) \uplus E(3)$. Nous avons alors dans $\mathcal{F}(E)$ la substitution partielle suivante :

$$(3.1.32)$$

L'*opérade libre* sur E est la symétrisée de l'opérade ns libre $\mathcal{F}(E)$. Nous la notons également $\mathcal{F}(E)$. Un élément de cette opérade est donc un arbre muni d'une permutation qui spécifie un ordre total sur ses feuilles.

Présentations par générateurs et relations

L'intérêt principal d'introduire les opérades libres provient du fait que toute opérade peut être définie par l'intermédiaire de *générateurs* engendrant une opérade libre et de *relations* qui définissent un quotient de cette dernière. En effet, toute opérade \mathcal{P} admet l'écriture

$$\mathcal{P} = \mathcal{F}(E)/_R, \quad (3.1.33)$$

où $E := \uplus_{n \geq 1} E(n)$ est l'ensemble des générateurs de \mathcal{P} et R est un idéal de $\mathcal{F}(E)$, l'espace des relations de \mathcal{P} . L'écriture de \mathcal{P} sous la forme (3.1.33) forme une *présentation* de \mathcal{P} .

Un élément x d'une opérade de présentation $\mathcal{F}(E)/_R$ admet l'entier n pour *degré* s'il est possible d'obtenir x par des substitutions partielles impliquant n occurrences de générateurs de E .

Une opérade \mathcal{P} est *binnaire* si elle admet une présentation $\mathcal{F}(E)/_R$ où $\#E(2) \neq 0$ et $\#E(n) = 0$ pour tout $n \neq 2$. Une opérade \mathcal{P} est *quadratique* si elle admet une présentation $\mathcal{F}(E)/_R$ où R est engendré uniquement par des éléments de degré deux de $\mathcal{F}(E)$.

Un grand nombre d'opérades connues sont à la fois binaires et quadratiques. Elles sont ainsi générées par des éléments d'arité deux et sont soumises à des relations engendrées par des éléments de degré deux.

3.1.3 Algèbres sur une opérade

L'un des principaux points forts de la théorie des opérades est que la plupart des définitions des structures algébriques peuvent être encodées et manipulées par l'intermédiaire d'opérades. Par exemple, les algèbres associatives, les algèbres associatives et commutatives, les algèbres de Lie, pour ne citer que ces exemples, sont chacune d'elles gouvernées par une opérade bien précise. De plus, il est intéressant de constater que l'inverse reste vrai : à toute opérade est associé un type d'algèbre. Nous décrivons justement ici comment obtenir une algèbre à partir d'une opérade.

L'opérade des applications linéaires

Nous avons besoin avant tout de la définition d'une opérade particulière. Soit V un espace vectoriel. L'*opérade des applications linéaires* de V , notée App_V , est définie comme suit. Les éléments de $App_V(n)$ sont les applications linéaires de la forme $V^{\otimes n} \rightarrow V$. Pour tous $x \in App_V(n)$, $y \in App_V(m)$ et $i \in [n]$, la substitution partielle $x \circ_i y$ est l'application linéaire $V^{\otimes n+m-1} \rightarrow V$ qui vérifie, pour tout $a_1 \otimes \cdots \otimes a_{n+m-1} \in V^{\otimes n+m-1}$,

$$(x \circ_i y)(a_1 \otimes \cdots \otimes a_{n+m-1}) = x(a_1 \otimes \cdots \otimes a_{i-1} \otimes y(a_i \otimes \cdots \otimes a_{i+m-1}) \otimes a_{i+m} \otimes \cdots \otimes a_{n+m-1}). \quad (3.1.34)$$

L'unité $\mathbf{1}$ de App_V est l'application identité sur V et l'action du groupe symétrique sur App_V vérifie, pour tous $x \in App_V(n)$, $\sigma \in \mathfrak{S}_n$ et $a_1 \otimes \cdots \otimes a_n \in V$,

$$(x \cdot \sigma)(a_1 \otimes \cdots \otimes a_n) = x(a_{\sigma_1^{-1}} \otimes \cdots \otimes a_{\sigma_n^{-1}}). \quad (3.1.35)$$

Algèbres sur une opérade

Soit \mathcal{P} une opérade et V un espace vectoriel. L'espace vectoriel V est muni d'une structure d'*algèbre sur l'opérade* \mathcal{P} s'il existe un morphisme d'opérade $\phi : \mathcal{P} \rightarrow App_V$. L'application ϕ permet de définir des opérations sur V de la manière suivante. Soit pour tout $n \geq 1$ l'application d'*évaluation*

$$ev : \mathcal{P}(n) \otimes V^{\otimes n} \rightarrow V, \quad (3.1.36)$$

définie linéairement pour tous $x \in \mathcal{P}(n)$, $a_1 \otimes \cdots \otimes a_n \in V$ par

$$ev(x, a_1 \otimes \cdots \otimes a_n) := \phi(x)(a_1 \otimes \cdots \otimes a_n). \quad (3.1.37)$$

L'application ev permet de calculer un élément de V en sortie sur l'entrée d'un élément de \mathcal{P} d'arité n et de n éléments de V . Ainsi, à chaque élément de \mathcal{P} correspond une loi de composition sur V .

3.2 Exemples d'opérades

Illustrons les notions rappelées dans le paragraphe précédent à travers quatre exemples d'opérades célèbres.

3.2.1 L'opérade commutative associative

Soit $Com := \bigoplus_{n \geq 1} Com(n)$ un espace vectoriel gradué tel que $Com(n) := \text{Vect}(\{x_n\})$ pour tout $n \geq 1$. On munit Com de la substitution partielle \circ_i définie linéairement par

$$x_n \circ_i x_m := x_{n+m-1}, \quad (3.2.1)$$

pour tous $n, m \geq 1$ et $i \in [n]$. Les espaces vectoriels $Com(n)$ sont également munis de l'action triviale du groupe symétrique, *i.e.*, $x_n \cdot \sigma = x_n$ pour tous $n \geq 1$ et $\sigma \in \mathfrak{S}_n$. Avec ces données, Com est une opérade ensembliste : c'est l'*opérade commutative associative*.

D'après la définition de Com , sa série de Hilbert est

$$F_{Com}^e(t) = \exp(t) - 1 = t + \frac{1}{2}t^2 + \frac{1}{6}t^3 + \frac{1}{24}t^4 + \cdots. \quad (3.2.2)$$

Il est de plus immédiat que Com est engendré par l'élément x_2 . Par conséquent, pour munir un espace vectoriel V d'une structure d'algèbre sur Com , et donc de manière équivalente, pour définir un morphisme d'opérade $\phi : Com \rightarrow App_V$, il suffit de définir l'image de x_2 par ϕ . Il est de plus immédiat que l'opérateur qui correspond à x_2 est commutatif et associatif. Ainsi, toute algèbre sur Com est commutative et associative.

3.2.2 L'opérade associative

L'opérade associative Ass est la symétrisée de la version non symétrique de Com . En d'autres termes, nous avons, étant donnée que $\dim Com(n) = 1$,

$$Ass(n) = Com(n) \otimes Vect(\mathfrak{S}_n) \simeq Vect(\mathfrak{S}_n), \quad (3.2.3)$$

pour tout $n \geq 1$, ce qui montre au passage que les éléments de Ass peuvent être identifiés à des permutations. La substitution partielle de Ass vérifie $\sigma \circ_i \nu = B_i(\sigma, \nu)$ pour tous $\sigma \in Ass(n)$, $\nu \in Ass(m)$ et $i \in [n]$. L'action du groupe symétrique sur Ass est simplement la composition des permutations. Sa série de Hilbert est

$$F_{Ass}^e(t) = \frac{t}{1-t} = t + t^2 + t^3 + t^4 + \dots \quad (3.2.4)$$

Remarquons que Ass est une opérade ensembliste et qu'elle est engendrée par la permutation $\sigma := 12$. Il est en effet évident que tout élément de Ass peut être obtenu en substituant σ avec lui-même et en appliquant ensuite l'action d'une permutation appropriée. Ainsi, toute algèbre sur Ass est munie d'un unique opérateur qui correspond à σ et qui est associatif.

Nous pouvons également définir cette opérade en donnant sa présentation. Son ensemble de générateurs E est de la forme

$$E := \emptyset \uplus \{\alpha\} \uplus \emptyset \uplus \dots, \quad (3.2.5)$$

et son espace des relations R est un sous-espace vectoriel de $\mathcal{F}(E)$ de la forme

$$R := 0 \oplus 0 \oplus R(3) \oplus 0 \oplus \dots, \quad (3.2.6)$$

où $R(3)$ est engendré par l'élément

$$\alpha \circ_1 \alpha - \alpha \circ_2 \alpha. \quad (3.2.7)$$

L'élément (3.2.7) encode l'associativité du générateur α . L'élément α correspond bien entendu à l'élément σ de la définition précédente de Ass et nous avons $Ass = \mathcal{F}(E)/R$. Ainsi, d'après sa présentation, Ass est à la fois binaire et quadratique.

3.2.3 L'opérade de Lie

Soit E un ensemble de générateurs de la forme

$$E := \emptyset \uplus \{\beta\} \uplus \emptyset \uplus \dots, \quad (3.2.8)$$

et R un sous-espace vectoriel de $\mathcal{F}(E)$ de la forme

$$R := 0 \oplus R(2) \oplus R(3) \oplus 0 \oplus \dots, \quad (3.2.9)$$

où $R(2)$ est engendré par l'élément

$$\beta + (\beta \cdot 21), \quad (3.2.10)$$

et $R(3)$ est engendré par l'élément

$$(\beta \circ_2 \beta) + (\beta \circ_2 \beta) \cdot \sigma + (\beta \circ_2 \beta) \cdot \sigma^2, \quad (3.2.11)$$

où $\sigma := 231$. La relation (3.2.10) encode l'antisymétrie de l'opérateur β et la relation (3.2.11) est connue sous le nom d'*identité de Jacobi*. L'opérade $Lie := \mathcal{F}(E)/R$ est l'opérade de Lie. Cette opérade est une sous-opérade de Ass puisque l'on peut voir Lie comme la sous-opérade de Ass engendrée par l'élément $\alpha - (\alpha \cdot 21)$ (voir [AL07]). De plus, sa série de Hilbert est de la forme

$$F_{Lie}^e(t) = -\log(1-t) = t + \frac{1}{2}t^2 + \frac{1}{3}t^3 + \frac{1}{4}t^4 + \dots \quad (3.2.12)$$

Une algèbre sur l'opéride *Lie* est une *algèbre de Lie*, *i.e.*, un espace vectoriel V sur lequel est défini une application linéaire $[-, -] : V \otimes V \rightarrow V$ appelée *crochet de Lie*, qui vérifie pour tous $x, y, z \in V$ les relations

$$[x, y] + [y, x] = 0, \quad (3.2.13)$$

et

$$[x, [y, z]] + [y, [z, x]] + [z, [x, y]] = 0. \quad (3.2.14)$$

Notons la correspondance entre (3.2.10) et (3.2.13), et entre (3.2.11) et (3.2.14).

3.2.4 L'opéride dendriforme

Soit E un ensemble de générateurs de la forme

$$E := \emptyset \uplus \{<, >\} \uplus \emptyset \uplus \dots, \quad (3.2.15)$$

et R un sous-espace vectoriel de $\mathcal{F}(E)$ de la forme

$$R := 0 \oplus 0 \oplus R(3) \oplus 0 \oplus \dots, \quad (3.2.16)$$

où $R(3)$ est engendré par les éléments

$$< \circ_1 < - < \circ_2 < - < \circ_2 >, \quad (3.2.17)$$

$$> \circ_2 < - < \circ_1 >, \quad (3.2.18)$$

$$> \circ_2 > - > \circ_1 > - > \circ_1 <. \quad (3.2.19)$$

L'opéride ns $Dend := \mathcal{F}(E)/R$ est l'*opéride dendriforme*. Sa série de Hilbert est de la forme

$$F_{Dend}(t) = \frac{1 - \sqrt{1 - 4t}}{2t} = t + 2t^2 + 5t^3 + 14t^4 + 42t^5 + 132t^6 + 429t^7 + \dots, \quad (3.2.20)$$

et est la série génératrice des objets catalans.

Cette opéride fut introduite par Loday [Lod01]. Les algèbres sur $Dend$ sont les *algèbres dendriformes* (voir le paragraphe 5.2.4 du chapitre 5 pour plus de détails) et il est remarquable que l'espace vectoriel sous-jacent à l'algèbre dendriforme libre sur un générateur est l'espace vectoriel combinatoire engendré par les arbres binaires [LR02] (voir aussi [LR98] et [LV10]).

Remarquons au passage que l'opérateur $\gamma := < + >$ est associatif, *i.e.*, il vérifie la relation $\gamma \circ_1 \gamma = \gamma \circ_2 \gamma$. En effet, en utilisant les relations (3.2.17), (3.2.18) et (3.2.19), nous avons

$$(< + >) \circ_1 (< + >) = \textcolor{red}{<} \textcolor{red}{\circ_1} \textcolor{red}{<} + \textcolor{blue}{<} \textcolor{blue}{\circ_1} \textcolor{blue}{>} + \textcolor{green}{>} \textcolor{green}{\circ_1} \textcolor{green}{<} + \textcolor{green}{>} \textcolor{green}{\circ_1} \textcolor{green}{>} \quad (3.2.21)$$

$$= \textcolor{red}{<} \textcolor{red}{\circ_2} \textcolor{red}{<} + \textcolor{red}{<} \textcolor{red}{\circ_2} \textcolor{red}{>} + \textcolor{blue}{>} \textcolor{blue}{\circ_2} \textcolor{blue}{<} + \textcolor{blue}{>} \textcolor{blue}{\circ_2} \textcolor{blue}{>} \quad (3.2.22)$$

$$= (< + >) \circ_2 (< + >). \quad (3.2.23)$$

3.3 Groupes et algèbres de Hopf associés à une opéride

Nous rappelons dans ce paragraphe une construction qui à une opéride associe un groupe, une algèbre de Hopf commutative et une algèbre de Hopf non commutative. Ces constructions possèdent diverses applications. L'une est de nature combinatoire : le groupe associé à l'opéride dendriforme apparaît comme un outil fondamental pour compter le nombre d'intervalles dans le treillis de Tamari [Cha06a] ; l'autre, de nature plus algébrique : l'algèbre de Hopf commutative associée à l'opéride non associative permutative *NAP* [Liv06] permet de retrouver, en tant que quotient, l'algèbre de Hopf de Connes-Kreimer [CL07].

3.3.1 Groupe associé à une opérade

La construction que nous rappelons ici, considérée en [KM01], [Cha02], [vdL04], [CL07] et [Fra08] (voir aussi [Cha08] et [LV10]), associe à toute opérade un monoïde de séries formelles sur ses éléments. Un groupe est ensuite obtenu en se restreignant à ses éléments inversibles.

Soit $(\mathcal{P}, \circ, \mathbf{1})$ une opérade. Si \mathcal{P} est symétrique, nous considérons sa version non symétrique en oubliant l'action du groupe symétrique. Soit $\text{Ser}(\mathcal{P})$ l'ensemble des *séries formelles* d'éléments de \mathcal{P} , ou en d'autres termes,

$$\text{Ser}(\mathcal{P}) := \left\{ x := \sum_{n \geq 1} x^{(n)} : x^{(n)} \in \mathcal{P}(n) \right\}. \quad (3.3.1)$$

Sur cet ensemble, on définit un produit \cdot à partir de l'opérateur de substitution complète \circ de \mathcal{P} par

$$(x \cdot y)^{(n)} := \sum_{i_1 + \dots + i_k = n} x^{(k)} \circ [y^{(i_1)}, \dots, y^{(i_k)}], \quad (3.3.2)$$

pour tous $x, y \in \text{Ser}(\mathcal{P})$. La relation d'associativité de \circ implique l'associativité du produit défini en (3.3.2). De plus, étant donné que l'unité $\mathbf{1}$ de \mathcal{P} est aussi un élément de $\text{Ser}(\mathcal{P})$, cet élément est de manière évidente l'unité de ce produit. Ainsi, $(\text{Ser}(\mathcal{P}), \cdot, \mathbf{1})$ est un monoïde.

Considérons le sous-ensemble suivant de $\text{Ser}(\mathcal{P})$:

$$\text{Ser}_G(\mathcal{P}) := \left\{ x \in \text{Ser}(\mathcal{P}) : x^{(1)} = \lambda \mathbf{1} \right\}, \quad (3.3.3)$$

où $x^{(1)}$ est le résultat de la projection de x sur sa composante homogène constituée des éléments d'arité 1 et λ est un coefficient non nul de \mathbb{K} . Étant donné que le produit de deux éléments de $\text{Ser}_G(\mathcal{P})$ est de la forme

$$\left(\lambda_1 \mathbf{1} + \sum_{i \geq 2} x^{(i)} \right) \cdot \left(\lambda_2 \mathbf{1} + \sum_{j \geq 2} y^{(j)} \right) = \lambda_1 \lambda_2 \mathbf{1} + \sum_{n \geq 2} z^{(n)}, \quad (3.3.4)$$

l'ensemble $\text{Ser}_G(\mathcal{P})$ est un sous-monoïde de $\text{Ser}(\mathcal{P})$. En outre, ce sous-monoïde est un groupe puisque l'on peut calculer arité par arité l'inverse d'un élément. En effet, en posant

$$x^{-1} := \left(\lambda_1 \mathbf{1} + \sum_{i \geq 2} x^{(i)} \right)^{-1} = \left(\lambda_2 \mathbf{1} + \sum_{i \geq 2} y^{(i)} \right) =: y, \quad (3.3.5)$$

on trouve $\lambda_1 = \lambda_2 =: \lambda$ et, à titre d'exemple jusqu'en arité 3,

$$x^{(1)} \circ [y^{(1)}] = \mathbf{1}, \quad (3.3.6)$$

$$x^{(1)} \circ [y^{(2)}] + x^{(2)} \circ [y^{(1)}, y^{(1)}] = 0, \quad (3.3.7)$$

$$x^{(1)} \circ [y^{(3)}] + x^{(2)} \circ [y^{(1)}, y^{(2)}] + x^{(2)} \circ [y^{(2)}, y^{(1)}] + x^{(3)} \circ [y^{(1)}, y^{(1)}, y^{(1)}] = 0, \quad (3.3.8)$$

ce qui implique que les premières composantes homogènes de y sont

$$y^{(1)} = \frac{1}{\lambda} \mathbf{1}, \quad (3.3.9)$$

$$y^{(2)} = -\frac{1}{\lambda^3} x^{(2)}, \quad (3.3.10)$$

$$y^{(3)} = \frac{1}{\lambda^5} x^{(2)} \circ_1 x^{(2)} + \frac{1}{\lambda^5} x^{(2)} \circ_2 x^{(2)} - \frac{1}{\lambda^4} x^{(3)}. \quad (3.3.11)$$

Cette construction est en outre un foncteur de la catégorie des opérades ns (avec morphismes d'opérade) vers la catégorie des groupes (avec morphismes de groupe) [Cha02]. Le sous-groupe de $\text{Ser}_G(\mathcal{P})$ constitué des éléments dont le coefficient associé à $\mathbf{1}$ est $1_{\mathbb{K}}$ est le *sous-groupe affine* de $\text{Ser}_G(\mathcal{P})$ et est noté $\text{Ser}_G^1(\mathcal{P})$.

Donnons à présent un exemple. Soit \mathcal{P} l'opérade commutative associative Com dont la définition est rappelée dans le paragraphe 3.2.1 du chapitre 3. Nous avons dans $\text{Ser}_G(\mathcal{P})$ les produits

$$(x_1 + x_2) \cdot (x_1 + x_3) = x_1 + x_2 + x_3 + 2x_4 + x_6, \quad (3.3.12)$$

$$(x_1 + x_3) \cdot (x_1 + x_2) = x_1 + x_2 + x_3 + 3x_4 + 3x_5 + x_6. \quad (3.3.13)$$

On remarque qu'avec la transformation $x_n \mapsto t^n$, $\text{Ser}_G(\mathcal{P})$ est isomorphe au groupe des séries formelles en t où le produit est la composition des séries.

3.3.2 Algèbres de Hopf associées à une opérade

Un autre construction classique permet de construire une algèbre de Hopf à partir d'un groupe G : c'est l'*algèbre de Hopf des fonctions sur le groupe G* . Nous rappelons ici cette construction.

Construction commutative

Soit \mathcal{P} une opérade combinatoire ensembliste telle que $\dim \mathcal{P}(1) = 1$ et G le groupe $\text{Ser}_G^1(\mathcal{P})$. L'algèbre de Hopf $(\mathcal{H}_{\mathcal{P}}, \cdot, u, \Delta, v)$ des fonctions sur G peut être vue comme suit. En tant qu'espace vectoriel $\mathcal{H}_{\mathcal{P}}$ est engendré par les éléments de la forme

$$E_{\{x_1, \dots, x_n\}}, \quad (3.3.14)$$

où $\{x_1, \dots, x_n\}$ est un *multi-ensemble* d'éléments de \mathcal{P} . Le produit de $\mathcal{H}_{\mathcal{P}}$ est commutatif et est défini linéairement par

$$E_{\{x_1, \dots, x_n\}} \cdot E_{\{y_1, \dots, y_m\}} := E_{\{x_1, \dots, x_n, y_1, \dots, y_m\}}, \quad (3.3.15)$$

pour tous $x_1, \dots, x_n, y_1, \dots, y_m \in \mathcal{P}$ et l'unité u linéairement par

$$u(1_{\mathbb{K}}) := E_{\{\mathbf{1}\}} = E_{\emptyset} =: 1. \quad (3.3.16)$$

Le coproduit de Δ est défini linéairement par

$$\Delta(E_{\{x\}}) := \sum_{k \in [n]} \sum_{\substack{y \in \mathcal{P} \\ z_1, \dots, z_k \in \mathcal{P} \\ y \circ [z_1, \dots, z_k] = x}} E_{\{y\}} \otimes E_{\{z_1\}} \cdot \dots \cdot E_{\{z_k\}}, \quad (3.3.17)$$

pour tout $x \in \mathcal{P}(n)$. L'expression (3.3.17) définit un coproduit sur $\mathcal{H}_{\mathcal{P}}$ tout entier car Δ se doit d'être un morphisme d'algèbre et que, en tant qu'algèbre, $\mathcal{H}_{\mathcal{P}}$ est engendrée par les éléments de la forme $E_{\{x\}}$ où $x \in \mathcal{P}$. La coassociativité de Δ est impliquée par l'associativité de l'opérateur de substitution complète de \mathcal{P} . Finalement, la counité c de $\mathcal{H}_{\mathcal{P}}$ est définie linéairement par

$$c(E_{\{x_1, \dots, x_n\}}) := \delta_{\{x_1, \dots, x_n\}, \mathbf{1}}, \quad (3.3.18)$$

pour tous $x_1, \dots, x_n \in \mathcal{P}$. Nous appelons la base des E la *base élémentaire* de $\mathcal{H}_{\mathcal{P}}$. Remarquons que cette base est du côté gauche (voir le paragraphe 2.1.3 du chapitre 2).

D'après l'expression du produit (3.3.15), $\mathcal{H}_{\mathcal{P}}$ est libre en tant qu'algèbre commutative et ses éléments indécomposables sont de la forme $E_{\{x\}}$ où $x \in \mathcal{P}$. De plus, en posant $n - 1$ le degré d'un élément $E_{\{x\}}$ où $x \in \mathcal{P}(n)$, $\mathcal{H}_{\mathcal{P}}$ est une bigèbre graduée. Maintenant, étant donné que $\dim \mathcal{P}(1) = 1$, le seul élément de base de degré 0 est E_{\emptyset} , ce qui montre que $\mathcal{H}_{\mathcal{P}}$ est connexe, et implique qu'il s'agit d'une algèbre de Hopf combinatoire. Celle-ci est toujours commutative mais peut être cocommutative ou non cocommutative.

Nous avons de plus la caractérisation suivante pour la série de Hilbert de $\mathcal{H}_{\mathcal{P}}$:

Lemme 3.3.1. *La série de Hilbert $F_{\mathcal{H}_{\mathcal{P}}}(t)$ de l'algèbre de Hopf $\mathcal{H}_{\mathcal{P}}$ construite comme décrit ci-dessus vérifie*

$$F_{\mathcal{H}_{\mathcal{P}}}(t) = \exp \left(\sum_{n \geq 1} \frac{F_{\mathcal{P}}(t^n)}{nt^n} - \frac{1}{n} \right), \quad (3.3.19)$$

où \mathcal{P} est une opérade ensembliste telle que $\dim \mathcal{P}(1) = 1$ et $F_{\mathcal{P}}(t)$ est la série de Hilbert de la version non symétrique de \mathcal{P} .

Démonstration. La série génératrice d'une classe combinatoire C , dont les éléments sont des multi-ensembles d'éléments d'une autre classe combinatoire D qui ne possède pas d'élément de taille 0, s'exprime par (voir par exemple [FS09])

$$\mathcal{S}_C(t) = \exp \left(\sum_{n \geq 1} \frac{1}{n} \mathcal{S}_D(t^n) \right). \quad (3.3.20)$$

Comme $\mathcal{H}_{\mathcal{P}}$ est commutative, tout élément de base $E_{\{x_1, \dots, x_n\}}$ est indexé par un multi-ensemble d'éléments de \mathcal{P} . Le lemme provient du fait que $\mathcal{H}_{\mathcal{P}}$ est libre, et que, comme $\dim \mathcal{P}(1) = 1$ et qu'un générateur algébrique $E_{\{x\}}$ de $\mathcal{H}_{\mathcal{P}}$ est de degré $n - 1$ si $x \in \mathcal{P}(n)$, la série génératrice des générateurs algébriques de $\mathcal{H}_{\mathcal{P}}$ est de la forme $(F_{\mathcal{P}}(t) - t)/t$. \square

Un exemple de construction dans le cas commutatif

Donnons maintenant un exemple de construction avec l'opérade commutative associative Com vue comme une opérade ensembliste. On identifie ici tout élément $x_n \in Com(n)$ par l'entier $n - 1$. Ainsi avec cet encodage, $Com(n) = \{n - 1\}$ pour tout $n \geq 1$, et $n \circ_i m = n + m$. Les éléments de la forme $E_{\{n\}}$ avec $n \geq 1$ de \mathcal{H}_{Com} sont les générateurs algébriques de cette algèbre et sa base élémentaire est donnée par les $E_{\{x_1, \dots, x_k\}}$ où $x_1, \dots, x_k \geq 1$. Étant donné que \mathcal{H}_{Com} est commutative par construction, on impose la condition $x_1 \geq \dots \geq x_k$. Ainsi, chaque élément de base de degré n de \mathcal{H}_{Com} est indexé par une suite décroissante d'entiers dont la somme de ses éléments est n . Cet objet combinatoire est connu sous le nom de *partition d'entiers*. La *taille* d'une partition d'entiers $\lambda := \lambda_1, \dots, \lambda_k$ est la somme $\lambda_1 + \dots + \lambda_k$ et est notée $|\lambda|$. Les lettres λ_i sont appelées *parts* de λ . La série de Hilbert de \mathcal{H}_{Com} est ainsi la série génératrice des partitions d'entiers, dont les coefficients forment la suite [A000041](#) de [Slo] et est de la forme

$$F_{\mathcal{H}_{Com}}(t) = 1 + t + 2t^2 + 3t^3 + 5t^4 + 7t^5 + 11t^6 + 15t^7 + 22t^8 + 30t^9 + \dots \quad (3.3.21)$$

Voyons à présent le coproduit de \mathcal{H}_{Com} . On commence par le calculer sur ses générateurs algébriques en base dimension :

$$\Delta(E_1) = 1 \otimes E_1 + E_1 \otimes 1, \quad (3.3.22)$$

$$\Delta(E_2) = 1 \otimes E_2 + 2E_1 \otimes E_1 + E_2 \otimes 1, \quad (3.3.23)$$

$$\Delta(E_3) = 1 \otimes E_3 + E_1 \otimes E_{1,1} + 2E_1 \otimes E_2 + 3E_2 \otimes E_1 + E_3 \otimes 1, \quad (3.3.24)$$

$$\begin{aligned} \Delta(E_4) = & 1 \otimes E_4 + 2E_1 \otimes E_3 + 2E_1 \otimes E_{2,1} + 3E_2 \otimes E_2 + 3E_2 \otimes E_{1,1} \\ & + 4E_3 \otimes E_1 + E_4 \otimes 1. \end{aligned} \quad (3.3.25)$$

En particulier, (3.3.24) et (3.3.25) montrent que \mathcal{H}_{Com} n'est pas cocommutative. D'après la définition du coproduit donnée en (3.3.17), la règle pour la substitution partielle dans Com et notre encodage des éléments de base de \mathcal{H}_{Com} , nous avons la caractérisation suivante pour le coproduit de \mathcal{H}_{Com} . Le coefficient associé à un tenseur $\mathbf{E}_k \otimes \mathbf{E}_\lambda$ dans le coproduit d'un élément \mathbf{E}_n où λ est une partition d'entiers telle que $|\lambda| + k = n$, est le nombre de façon de placer la totalité de ses parts dans $k + 1$ emplacements distinguables. Par exemple, le coefficient de $\mathbf{E}_2 \otimes \mathbf{E}_{2,1}$ dans $\Delta(\mathbf{E}_5)$ est 6 puisque l'on a les six configurations suivantes : $(2, 1, \emptyset)$, $(2, \emptyset, 1)$, $(\emptyset, 2, 1)$, $(1, 2, \emptyset)$, $(1, \emptyset, 2)$ et $(\emptyset, 1, 2)$.

L'algèbre \mathcal{H}_{Com} est l'algèbre de Hopf de Fàa di Bruno.

Construction non commutative

Une version non commutative de la construction précédente existe et est considérée par exemple en [LV10] du point de vue dual de celle que nous décrivons ici. Nous donnons en effet ici l'analogue non commutatif de l'algèbre de Hopf $\mathcal{H}_{\mathcal{P}}$.

Considérons que le produit défini en (3.3.15) est non commutatif et notons $\mathfrak{H}_{\mathcal{P}}$ l'algèbre libre non commutative engendrée par les éléments de la forme \mathbf{E}_x où $x \in \mathcal{P}$. Nous notons maintenant \mathbf{E} la *base élémentaire* de $\mathfrak{H}_{\mathcal{P}}$, analogue à la base élémentaire de $\mathcal{H}_{\mathcal{P}}$. Les éléments de base de $\mathfrak{H}_{\mathcal{P}}$ sont de la forme

$$\mathbf{E}_{x_1|\dots|x_n}, \quad (3.3.26)$$

où $x_1|\dots|x_n$ est un *mot* d'éléments de \mathcal{P} . Le produit de $\mathfrak{H}_{\mathcal{P}}$ est par définition non commutatif et vérifie

$$\mathbf{E}_{x_1|\dots|x_n} \cdot \mathbf{E}_{y_1|\dots|y_m} := \mathbf{E}_{x_1|\dots|x_n|y_1|\dots|y_m}, \quad (3.3.27)$$

pour tous $x_1, \dots, x_n, y_1, \dots, y_m \in \mathcal{P}$ et le coproduit de Δ vérifie

$$\Delta(\mathbf{E}_x) := \sum_{k \in [n]} \sum_{\substack{y \in \mathcal{P} \\ z_1, \dots, z_k \in \mathcal{P} \\ y \circ [z_1, \dots, z_k] = x}} \mathbf{E}_y \otimes \mathbf{E}_{z_1} \cdot \dots \cdot \mathbf{E}_{z_k}, \quad (3.3.28)$$

Les éléments indécomposables de $\mathfrak{H}_{\mathcal{P}}$ sont de la forme \mathbf{E}_x où $x \in \mathcal{P}$. Sous réserve qu'il y ait dans \mathcal{P} au moins deux éléments, $\mathfrak{H}_{\mathcal{P}}$ est non commutative mais peut être cocommutative ou non cocommutative.

Nous avons de plus la caractérisation suivante pour la série de Hilbert de $\mathfrak{H}_{\mathcal{P}}$:

Lemme 3.3.2. *La série de Hilbert $F_{\mathfrak{H}_{\mathcal{P}}}(t)$ de l'algèbre de Hopf $\mathfrak{H}_{\mathcal{P}}$ construite comme décrit ci-dessus vérifie*

$$F_{\mathfrak{H}_{\mathcal{P}}}(t) = \frac{t}{2t - F_{\mathcal{P}}(t)}, \quad (3.3.29)$$

où \mathcal{P} est une opérade ensembliste telle que $\dim \mathcal{P}(1) = 1$ et $F_{\mathcal{P}}(t)$ est la série de Hilbert de la version non symétrique de \mathcal{P} .

Démonstration. La série génératrice d'une classe combinatoire C , dont les éléments sont des suites finies d'éléments d'une autre classe combinatoire D qui ne possède pas d'élément de taille 0, s'exprime par

$$\mathcal{S}_C(t) = \frac{1}{1 - \mathcal{S}_D(t)}. \quad (3.3.30)$$

Comme $\mathfrak{H}_{\mathcal{P}}$ est non commutative, tout élément de base $\mathbf{E}_{x_1|\dots|x_n}$ est indexé par une suite finie d'éléments de \mathcal{P} . Le lemme provient du fait que $\mathfrak{H}_{\mathcal{P}}$ est libre, et que, comme $\dim \mathcal{P}(1) = 1$ et qu'un générateur algébrique \mathbf{E}_x de $\mathfrak{H}_{\mathcal{P}}$ est de degré $n - 1$ si $x \in \mathcal{P}(n)$, la série génératrice des générateurs algébriques de $\mathfrak{H}_{\mathcal{P}}$ est de la forme $(F_{\mathcal{P}}(t) - t)/t$. \square

Deuxième partie

Combinatoire des couples d'arbres binaires jumeaux

Chapitre 4

Le monoïde de Baxter

Il existe une large gamme de monoïdes définis en tant que quotients du monoïde libre et dont les classes d'équivalence sont indexées par des objets combinatoires. Le premier exemple historique et sans doute le plus fondamental, est le monoïde plaxique [LS81], [Lot02]. Une définition de ce monoïde repose sur la donnée d'une congruence sur les mots obtenue à partir des relations de Knuth [Knu70]. Une autre approche, plus algorithmique, consiste à partir de l'algorithme de Schensted [Sch61], qui permet de déterminer sur l'entrée d'un mot, la longueur de son plus long sous-mot croissant. Cet algorithme construit un effet un tableau de Young à partir d'un mot, et il est naturel de considérer que deux mots sont équivalents s'ils produisent le même tableau lorsqu'on les insère. Il s'avère que ces deux approches mènent à la construction du même monoïde, que les éléments de ce dernier peuvent s'interpréter comme des tableaux de Young et que le produit du monoïde se calcule par l'intermédiaire de l'algorithme de Schensted. La principale illustration de l'importance du monoïde plaxique est son intervention dans la démonstration de la règle de Littlewood-Richardson [LR34] pour calculer les coefficients qui apparaissent lors de la multiplication de deux fonctions de Schur [Mac95]. Il permet de plus de construire une généralisation de l'algèbre des fonctions symétriques [Mac95] en la plongeant dans l'algèbre des tableaux de Young, rendant plus limpides certaines propriétés des fonctions symétriques.

Plus récemment, d'autres monoïdes qui ont des caractéristiques semblables au monoïde plaxique ont été introduits et étudiés. L'exemple le plus convaincant est dans doute le monoïde sylvestre [HNT02], [HNT05], qui, tout comme le monoïde plaxique, peut être défini de deux manières équivalentes, soit par une congruence sur les mots, soit par un algorithme d'insertion. Dans ce cas, c'est l'algorithme d'insertion dans un arbre binaire de recherche [AU94], [Knu98], [CLRS03] qui joue le rôle de l'algorithme de Schensted, et les objets du monoïde sont des arbres binaires de recherche. De plus, tout comme dans le cas plaxique, le monoïde sylvestre est un tremplin pour la construction d'une algèbre qui généralise celle des fonctions symétriques [LR98], [HNT05]. Notons que sous certaines conditions, de tels monoïdes mènent à la construction de treillis. Par exemple, le monoïde sylvestre offre une construction très intéressante [HNT05] du treillis de Tamari [Tam62], [HT72].

L'objectif de ce chapitre est d'introduire et d'étudier un monoïde semblable au monoïde plaxique et au monoïde sylvestre, dont les classes d'équivalence sont en bijection avec les objets de la famille combinatoire de Baxter. La famille combinatoire de Baxter est constituée des classes combinatoires en bijection avec les permutations de Baxter [Bax64], qui sont des permutations qui évitent certains motifs. Citons par exemple les couples d'arbres binaires jumeaux [DG94], les quadrangulations [ABP04] et les orientations planes bipolaires [BBMF08], qui sont des objets de cette famille. Nous considérerons spécialement les couples d'arbres binaires jumeaux comme représentants de cette famille.

Ce chapitre est organisé comme suit. Dans le paragraphe 4.1, nous présentons les concepts de base sur les monoïdes semblables au monoïde plaxique. Nous passons ensuite en revue quelques exemples de tels monoïdes : le monoïde plaxique, le monoïde sylvestre, le monoïde de Bell, et enfin, le monoïde des k -reculs. La définition du monoïde de Baxter est donnée dans le paragraphe 4.2. Nous y dégageons ses premières propriétés, ainsi que les liens qu'il entretient avec certains des monoïdes sus-cités. Le paragraphe 4.3 est consacrée à construire un analogue de l'algorithme d'insertion et de la correspondance de Robinson-Schensted. Nous associons ainsi bijectivement à tout mot, un couple de couples d'arbres binaires jumeaux étiquetés soumis à certaines conditions. Cette correspondance fait intervenir deux algorithmes d'insertion dans un arbre binaire de recherche, ainsi que la combinatoire des arbres binaires croissants et décroissants. De plus, à partir d'un couple d'arbres binaires jumeaux étiqueté J , nous proposons divers algorithmes d'extraction pour calculer un mot particulier dans la classe encodée par J . Nous terminons par le paragraphe 4.4 en rappelant la construction qui à une congruence d'un treillis associe un treillis quotient. La congruence de Baxter est, comme nous le montrons, une congruence du permutoèdre. Étant donné que les classes d'équivalence de permutations sous la congruence de Baxter sont en bijection avec les couples d'arbres binaires jumeaux non étiquetés, la congruence de Baxter définit un treillis quotient du permutoèdre sur l'ensemble de ces éléments. Nous décrivons ses relations de couverture et montrons que ce treillis est très similaire au treillis de Tamari. Un nouvel objet de la famille combinatoire de Baxter est finalement introduit, à savoir les *diagrammes de Tamari doubles*, dans le but de pouvoir comparer facilement deux éléments dans ce treillis.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir11].

4.1 Bons monoïdes

Dans cette partie, avant de donner des exemples des monoïdes connus et qui sont à bien des rapports analogues au monoïde plaxique, nous rappelons la définition de ce qu'est un *bon monoïde* [Hiv04], à comprendre au sens « semblable au monoïde plaxique ». Nous terminons cette partie en présentant des opérations sur les bons monoïdes qui permettent d'en produire de nouveaux à partir d'anciens.

4.1.1 Définitions de base

Définition 4.1.1. *Une relation d'équivalence \equiv définie sur A^* est une congruence de monoïde — ou simplement congruence si le contexte est clair — si pour tous mots $u, u', v, v' \in A^*$,*

$$u \equiv u' \quad \text{et} \quad v \equiv v' \quad \text{impliquent} \quad u \cdot v \equiv u' \cdot v'. \quad (4.1.1)$$

Une congruence \equiv permet de définir une structure de *monoïde quotient* du monoïde libre, que nous notons A^*/\equiv . En effet, si $\tau : A^* \rightarrow A^*/\equiv$ est la projection canonique, l'ensemble A^*/\equiv est muni du produit \cdot défini, pour toutes classes d'équivalence \hat{x} et \hat{y} de A^*/\equiv , par

$$\hat{x} \cdot \hat{y} := \tau(x \cdot y), \quad (4.1.2)$$

où x et y sont des éléments quelconques de \hat{x} et de \hat{y} respectivement.

Définition 4.1.2. *Une relation d'équivalence \equiv définie sur A^* est compatible avec la déstandardisation si pour tous mots $u, v \in A^*$,*

$$u \equiv v \quad \text{si et seulement si} \quad \text{std}(u) \equiv \text{std}(v) \quad \text{et} \quad \text{ev}(u) = \text{ev}(v). \quad (4.1.3)$$

Lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est *compatible avec la déstandardisation* si \equiv l'est.

Cette définition signifie en particulier que tous les mots d'une classe d'équivalence d'une relation compatible avec la déstandardisation possèdent la même évaluation, et à plus forte raison, la même longueur. Remarquons ainsi que la classe d'équivalence d'une permutation est constituée uniquement de permutations. En plus de cela, ce type de relation d'équivalence possède l'attrait supplémentaire d'être plus simple à étudier. On peut en effet restreindre l'étude de la relation aux mots sans répétition de lettres.

Définition 4.1.3. Une relation d'équivalence \equiv définie sur A^* est compatible aux restrictions aux intervalles d'alphabet si pour tout intervalle I de A et tous mots $u, v \in A^*$,

$$u \equiv v \quad \text{implique} \quad u|_I \equiv v|_I. \quad (4.1.4)$$

Lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est *compatible avec la restriction aux intervalles d'alphabet* si \equiv l'est.

La définition suivante, due à Hivert [Hiv04], met en évidence une classe de congruences sur les mots, et ainsi du même coup une classe de monoïdes quotients du monoïde libre :

Définition 4.1.4. Soit \equiv une relation d'équivalence sur A^* . Si \equiv est une congruence de monoïde, compatible avec la déstandardisation et aux restrictions aux intervalles d'alphabet, alors, le monoïde quotient A^*/\equiv est un bon monoïde.

Il est remarquable que tous les bons monoïdes connus sont définis par l'intermédiaire d'un ensemble R de relations binaires sur A^* . Une manière pratique de construire une congruence \equiv à partir d'un tel ensemble R consiste à considérer la relation binaire \leftrightarrow définie pour tout $u, v \in A^*$ par

$$u \leftrightarrow v \quad \text{s'il existe } \curvearrowright \in R \text{ tel que } u \curvearrowright v \text{ ou } v \curvearrowright u. \quad (4.1.5)$$

Deux mots sont déclarés équivalents s'ils sont en relation pour la congruence engendrée par la clôture réflexive et transitive de \leftrightarrow . En d'autres termes, on pose $xuy \equiv xu'y$ si $u = u'$ ou s'il existe une suite de mots $v^{(1)}, \dots, v^{(\ell)}$ tels que

$$u \leftrightarrow v^{(1)} \leftrightarrow \dots \leftrightarrow v^{(\ell)} \leftrightarrow u'. \quad (4.1.6)$$

Les relations de R sont appelées *relations d'adjacence* et peuvent dans certains cas être avantageusement vues comme des *règles de réécriture* — on pourra à ce sujet consulter [BN98]. Dans ce formalisme, deux mots sont équivalents si l'on peut réécrire l'un en l'autre par utilisation de la règle \leftrightarrow .

Dans ce qui suit, les congruences que nous allons considérer vont être définies en donnant simplement l'ensemble R des relations d'adjacence. Nous appelons alors la congruence ainsi obtenue à partir de R la *congruence engendrée par R* .

Plusieurs questions peuvent se poser lorsque l'on étudie un bon monoïde A^*/\equiv . L'une des plus naturelles porte sur l'encodage de ses classes d'équivalence et d'algorithmes efficaces pour décider si deux mots sont équivalents. Une autre question est d'étudier le comportement de \equiv lorsque celle-ci est restreinte aux permutations, par exemple en comptant combien il y a de classes de permutations de taille n . De plus, il est également naturel d'observer si les classes de permutations forment un intervalle du permutoèdre, ou mieux, si la congruence de monoïde \equiv est aussi une congruence de ce treillis.

Voici une autre question classique, dont nous rappelons la définition :

Définition 4.1.5. Une relation d'équivalence \equiv définie sur A^* est compatible avec l'involution de Schützenberger si pour tous mots $u, v \in A^*$,

$$u \equiv v \quad \text{si et seulement si} \quad u^\# \equiv v^\#. \quad (4.1.7)$$

Encore dans ce cas, lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est compatible avec l'involution de Schützenberger si \equiv l'est.

4.1.2 Quelques exemples de bons monoïdes

Nous donnons dans cette partie quelques exemples et propriétés de base de quelques bons monoïdes connus.

Le monoïde plaxique

Le *monoïde plaxique* [LS81], [Lot02] est le quotient de A^* par la congruence \equiv_P engendrée par les relations d'adjacence $\xrightarrow{P1}$ et $\xrightarrow{P2}$ définies pour $a, b, c \in A$ par

$$acb \xrightarrow{P1} cab \quad \text{où} \quad a \leq b < c, \quad (4.1.8)$$

$$bac \xrightarrow{P2} bca \quad \text{où} \quad a < b \leq c. \quad (4.1.9)$$

On vérifie sans peine que le monoïde plaxique est un bon monoïde. Il est de plus compatible avec l'involution de Schützenberger puisque l'image de la relation $\xrightarrow{P1}$ par l'involution de Schützenberger est $\xrightarrow{P2}$ et réciproquement.

Les classes plaxiques de permutations sont en bijection avec les tableaux de Young standard [LS81], [Ful97]. Cette bijection se montre en effet par l'intermédiaire de la correspondance de Robinson-Schensted. Ainsi, les premiers cardinaux des classes plaxiques de permutations par taille sont

$$1, 1, 2, 4, 10, 26, 76, 232, 764, 2620, 9496, \quad (4.1.10)$$

et comptent également le nombre d'involutions sur $[n]$ (suite [A000085](#) de [Slo]).

Le monoïde hypoplaxique

Le *monoïde hypoplaxique* est le quotient de A^* par la congruence \equiv_H engendrée par les relations d'adjacence $\xrightarrow{H1}$ et $\xrightarrow{H2}$ définies pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xrightarrow{H1} caub \quad \text{où} \quad a \leq b < c, \quad (4.1.11)$$

$$bua \xrightarrow{H2} bua \quad \text{où} \quad a < b \leq c. \quad (4.1.12)$$

Ce bon monoïde a été introduit par Krob et Thibon [KT97], [KT99] et étudié par Novelli [Nov98]. L'appellation « hypoplaxique » provient du fait qu'il forme un quotient du monoïde plaxique. En effet, pour tous mots u et v , si $u \equiv_P v$ alors $u \equiv_H v$.

Il est intéressant de noter que les classes de permutations sont en bijection avec les *mots binaires*, i.e., les mots sur l'alphabet $\{0, 1\}$. En effet, pour encoder une classe hypoplaxique de mots de taille n , il est nécessaire et suffisant d'encoder l'ensemble des reculs qui caractérisent cette classe. Toute classe hypoplaxique de permutations contient en effet toutes les permutations qui ont un ensemble de reculs donnés. Cet ensemble peut se représenter par le mot binaire b de longueur $n - 1$ où $b_i = 1$ si et seulement si i est un recul des éléments de la classe encodée. Les premiers cardinaux des classes hypoplaxiques de permutations par taille sont donc

$$1, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, \quad (4.1.13)$$

et comptent également le nombre de compositions de l'entier n (suite [A000079](#) de [Slo]).

Le monoïde sylvestre

Le *monoïde sylvestre* est le quotient de A^* par la congruence \equiv_s engendrée par la relation d'adjacence \xleftrightarrow{s} définie pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xleftrightarrow{s} caub \quad \text{où} \quad a \leq b < c. \quad (4.1.14)$$

Ce bon monoïde a été introduit par Hivert, Novelli et Thibon [HNT02], [HNT03], [HNT05]. Il possède un certain nombre de propriétés intéressantes, comme le fait que la congruence sylvestre est aussi une congruence de treillis du permutoèdre, et que le treillis obtenu par ce quotient est le treillis de Tamari [HNT05].

L'analogue sylvestre de l'algorithme d'insertion plaxique est l'algorithme d'insertion dans un arbre binaire de recherche droit. En effet, deux mots sont équivalents pour la relation sylvestre si et seulement si chacun d'eux donne le même arbre binaire de recherche droit lorsqu'on les insère de droite à gauche suivant cet algorithme [AU94], [Knu98], [CLRS03]. Ainsi, le \mathbb{P} -*symbole sylvestre* d'un mot est l'arbre binaire de recherche droit obtenu en insérant ses lettres de droite à gauche.

Introduisons une version légèrement modifiée de ce monoïde que nous appelons *monoïde #-sylvestre*. Ce monoïde est le quotient de A^* par la congruence $\equiv_{\#s}$, obtenue à partir de la relation d'adjacence $\xleftrightarrow{\#s}$ définie pour $u \in A^*$ et $a, b, c \in A$ par

$$bua \xleftrightarrow{\#s} buca \quad \text{où} \quad a < b \leq c. \quad (4.1.15)$$

Cette dénomination provient du fait que le monoïde #-sylvestre est défini à partir du monoïde sylvestre en considérant l'image de la relation d'adjacence sylvestre par l'involution de Schützenberger $\#$. En effet, pour tous mots $u, v \in A^*$, $u \equiv_{\#s} v$ si et seulement si $u^\# \equiv_s v^\#$.

Une version légèrement modifiée de l'algorithme d'insertion dans un arbre binaire de recherche permet également de décider de l'équivalence de deux mots sous la relation #-sylvestre. Dans notre contexte, nous appelons cet algorithme l'*insertion en feuille*, et celui-ci se décline donc en deux versions, l'une en faveur de la relation sylvestre, et l'autre, en faveur de la relation #-sylvestre.

Algorithme : INSERTIONFEUILLE.

Entrée : un arbre binaire de recherche droit (resp. gauche) T étiqueté sur A , une lettre $a \in A$.

Sortie : T après l'insertion en feuille de a .

1. Si $T = \perp$, retourner l'arbre binaire à un nœud étiqueté par a .
2. Soit b l'étiquette de la racine de T .
3. Si $a \leq b$ (resp. $a < b$) :
 - (a) alors, insérer en feuille récursivement a dans le sous-arbre gauche de T .
 - (b) Sinon, insérer en feuille récursivement a dans le sous-arbre droit de T .

Fin.

Dans le but d'y faire référence plus loin, rappelons ici le théorème de [HNT05] reformulé dans notre contexte, et complété pour qu'il reste valable dans le cadre de la relation #-sylvestre.

Théorème 4.1.6. *Deux mots sont équivalents pour la relation sylvestre (resp. #-sylvestre) si et seulement si chacun d'eux donne le même arbre binaire de recherche droit (resp. gauche) lorsque l'on insère ses lettres de droite à gauche (resp. gauche à droite).*

En d'autres termes, tout arbre binaire de recherche droit (resp. gauche) encode une classe d'équivalence sylvestre (resp. #-sylvestre), et réciproquement.

Expliquons la partie du théorème 4.1.6 qui concerne la relation #-sylvestre. Il s'ensuit de (4.1.15) qu'encoder une classe #-sylvestre d'un mot u est équivalent à encoder la classe sylvestre de $u^\#$. Pour cela, il suffit simplement d'insérer u de gauche à droite en considérant que la relation d'ordre portant sur ses lettres est inversée. De cette manière, nous obtenons un arbre binaire tel que pour chacun de ses nœuds x étiquetés par une lettre b , toutes les étiquettes a des nœuds du sous-arbre gauche de x , et toutes les étiquettes c des nœuds du sous-arbre droit de x , l'inégalité $a \geq b > c$ est établie. Cet arbre n'est clairement pas un arbre binaire de recherche gauche. Cependant, un arbre binaire de recherche gauche peut être obtenu en échangeant, pour chacun de ses nœuds, ses sous-arbres gauches et droits. Il est possible de montrer par récurrence sur $|u|$ que cet arbre binaire de recherche gauche est effectivement celui que produit INSERTIONFEUILLE en insérant les lettres de u de gauche à droite. Ceci justifie la différence de traitement entre arbres binaires de recherche gauches et droits au niveau de l'instruction (3) de INSERTIONFEUILLE.

Ainsi, comme il existe une unique manière d'étiqueter par une permutation un arbre binaire de manière à en faire un arbre binaire de recherche, les classes sylvestres et #-sylvestres de permutations de taille n peuvent s'encoder par des arbres binaires non étiquetés à n nœuds. Les premiers cardinaux des classes sylvestres de permutations par taille sont donc

$$1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, \quad (4.1.16)$$

et comptent les objets catalans (suite A000108 de [Slo]).

Le monoïde de Bell

Le *monoïde de Bell* est le quotient de A^* par la congruence \equiv_{BL} engendré par la relation d'adjacence \xleftrightarrow{BL} définie pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xleftrightarrow{BL} caub \quad \text{où} \quad a \leq b < c \text{ et pour tout } d \in \text{Alph}(u), d \geq c. \quad (4.1.17)$$

Ce bon monoïde a été introduit en [Rey07]. Un analogue de l'algorithme d'insertion de Robinson-Schensted y est construit, et permet de montrer que les classes de permutations de taille n sous la congruence \equiv_{BL} sont en bijection avec les partitions de l'ensemble $[n]$. Les premiers cardinaux des classes de permutations par taille sont

$$1, 1, 2, 5, 15, 52, 203, 877, 4140, 21147, 115975, \quad (4.1.18)$$

et comptent les partitions de l'ensemble $[n]$ (suite A000110 de [Slo]).

Nous pouvons observer que la relation \equiv_{BL} est un raffinement de la relation sylvestre car $u \equiv_{BL} v$ implique $u \equiv_S v$ pour tout mots u et v . De plus, toute classe de permutations de \equiv_{BL} est un intervalle du permutoèdre [Rey07].

Le monoïde des k -reculs

Si a et c sont deux lettres, on note par $c - a$ le cardinal de l'ensemble $\{b \in A : a < b \leq c\}$. Dans [NRT09], Novelli, Reutenauer et Thibon définissent, pour tout $k \geq 0$, la congruence \equiv_{kR} engendrée par la relation d'adjacence \xleftrightarrow{kR} définie pour $a, c \in A$ par

$$ac \xleftrightarrow{kR} ca \quad \text{où} \quad c - a \geq k. \quad (4.1.19)$$

Cette congruence définit le *monoïde des k -reculs*, qui n'est cependant pas en l'état un bon monoïde. En effet, ce monoïde n'est pas compatible avec la déstandardisation puisque l'on a par exemple $13 \equiv_{2R} 31$ mais $12 = \text{std}(13)$ n'est pas équivalent à $21 = \text{std}(31)$.

En reformulant la relation d'adjacence \xleftrightarrow{kR} pour $u, v \in A^*$ et $a, c \in A$ par

$$u \mathbf{a} c v \xleftrightarrow{kR} u \mathbf{c} a v \quad \text{où} \quad |u|_{[a,c]} + |v|_{[a,c]} \geq k - 1, \quad (4.1.20)$$

on obtient un bon monoïde. En d'autres termes, (4.1.20) signifie que l'on peut réécrire un mot w en transposant deux lettres adjacentes si et seulement s'il existe dans w au moins $k - 1$ lettres qui se trouvent en valeur entre a et c . Dans le reste du texte, nous considérerons cette reformulation lorsque nous ferons référence au monoïde des k -reculs.

Remarquons que la congruence \equiv_{2R} est la congruence hypoplaxique et que pour tout $k \geq 0$, la relation \equiv_{k+1R} est un raffinement de \equiv_{kR} , et ainsi, le monoïde des k -reculs peut être vu comme une généralisation du monoïde hypoplaxique.

4.1.3 Opérations sur les bons monoïdes

Nous définissons dans cette partie deux opérations sur les bons monoïdes qui permettent de construire de nouveaux bons monoïdes à partir d'anciens.

Définition 4.1.7. Soient M et N deux bons monoïdes définis respectivement à partir des congruences \equiv_M et \equiv_N . Soit \equiv la relation d'équivalence définie pour tous $u, v \in A^*$ par

$$u \equiv v \quad \text{si et seulement si} \quad u \equiv_M v \quad \text{et} \quad u \equiv_N v. \quad (4.1.21)$$

L l'intersection de M et de N est le monoïde A^*/\equiv .

Proposition 4.1.8. L'intersection de deux bons monoïdes est un bon monoïde.

Démonstration. Soient $M := A^*/\equiv_M$ et $N := A^*/\equiv_N$ deux bons monoïdes, et $L := A^*/\equiv$ la structure obtenue par l'intersection de M et de N . La démonstration du fait que la relation \equiv définie en (4.1.21) est bien une relation d'équivalence ainsi qu'une congruence est directe. Par conséquent, la définition 4.1.7 est consistante puisque L est bien un monoïde.

Montrons maintenant que L est bien un bon monoïde. Soient I un intervalle de A et $u, v \in A^*$ tels que $u \equiv v$. Par définition, $u \equiv_M v$ et $u \equiv_N v$, et comme \equiv_M et \equiv_N sont compatibles aux restrictions aux intervalles d'alphabet, $u|_I \equiv_M v|_I$ et $u|_I \equiv_N v|_I$. Par conséquent, nous avons $u|_I \equiv v|_I$, et \equiv est compatible aux restrictions aux intervalles d'alphabet. De plus, comme \equiv_M et \equiv_N sont compatibles avec la déstandardisation, nous avons $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv_M \text{std}(v)$ et $\text{std}(u) \equiv_N \text{std}(v)$. Ceci implique donc $\text{std}(u) \equiv \text{std}(v)$. Inversement, si u et v sont tels que $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv \text{std}(v)$, alors par définition de \equiv , $\text{std}(u) \equiv_M \text{std}(v)$ et $\text{std}(u) \equiv_N \text{std}(v)$. La compatibilité avec la déstandardisation de \equiv_M et \equiv_N implique $u \equiv_M v$ et $u \equiv_N v$, ce qui entraîne $u \equiv v$ et montre que \equiv est compatible avec la déstandardisation. \square

Définition 4.1.9. Soit M un bon monoïde défini à partir de la congruence \equiv_M . Soit \equiv la relation d'équivalence définie pour tous $u, v \in A^*$ par

$$u \equiv v \quad \text{si et seulement si} \quad u^\# \equiv_M v^\#. \quad (4.1.22)$$

L l'image par l'involution de Schützenberger de M est le monoïde A^*/\equiv .

Avant de montrer que l'image par l'involution de Schützenberger d'un bon monoïde est un bon monoïde, nous avons besoin du lemme suivant, qui souligne une forme de compatibilité entre la standardisation et l'involution de Schützenberger sur les mots :

Lemme 4.1.10. L'involution de Schützenberger et la standardisation commutent. Autrement dit, pour tout $u \in A^*$, $\text{std}(u^\#) = \text{std}(u)^\#$.

Démonstration. Par définition du processus de standardisation, la permutation $\sigma := \text{std}(u)$ est l'unique permutation de taille $n := |u|$ qui possède le même ensemble d'inversions que u . De plus, par définition de l'involution de Schützenberger, (i, j) est une inversion de σ si et seulement si $(n+1-j, n+1-i)$ est une inversion de $\sigma^\#$. De même, on observe que (i, j) est une inversion de u si et seulement si $(n+1-j, n+1-i)$ est une inversion de $u^\#$. Le résultat suit du fait qu'une permutation est entièrement déterminée par son ensemble d'inversions. \square

Proposition 4.1.11. *L'image par l'involution de Schützenberger d'un bon monoïde est un bon monoïde.*

Démonstration. Soient $M := A^*/\equiv_{\mathfrak{M}}$ un bon monoïde et $N := A^*/\equiv$ l'image par l'involution de Schützenberger de M . La démonstration du fait que la relation \equiv définie en (4.1.22) est bien une relation d'équivalence est directe.

Montrons que \equiv est bien une congruence de monoïdes. Soient u, u', v et v' des mots tels que $u \equiv u'$ et $v \equiv v'$. Nous avons alors $u^\# \equiv_{\mathfrak{M}} u'^\#$ et $v^\# \equiv_{\mathfrak{M}} v'^\#$. Comme $\equiv_{\mathfrak{M}}$ est une congruence, on a

$$u^\# \cdot v^\# \equiv_{\mathfrak{M}} u'^\# \cdot v'^\#, \quad (4.1.23)$$

et ainsi,

$$(u^\# \cdot v^\#)^\# \equiv (u'^\# \cdot v'^\#)^\#, \quad (4.1.24)$$

ce qui implique $v \cdot u \equiv v' \cdot u'$ par définition de l'involution de Schützenberger, et montre que \equiv est une congruence. La définition 4.1.9 est donc consistante puisque N est bien un monoïde.

Montrons maintenant que N est un bon monoïde. Soit I un intervalle de A et supposons $u \equiv v$. Alors, $u^\# \equiv_{\mathfrak{M}} v^\#$, et comme $\equiv_{\mathfrak{M}}$ est compatible aux restrictions aux intervalles d'alphabet, $(u^\#)_{|I} \equiv_{\mathfrak{M}} (v^\#)_{|I}$. Maintenant, comme on a $(w^\#)_{|E} = (w_{|E^\#})^\#$ pour tout mot w et tout alphabet E , nous avons $(u_{|I^\#})^\# \equiv_{\mathfrak{M}} (v_{|I^\#})^\#$, et ainsi, $u_{|I^\#} \equiv v_{|I^\#}$. Étant donné que $I^\#$ est également un intervalle de A , \equiv est compatible aux restrictions aux intervalles d'alphabet.

Supposons maintenant $u \equiv v$. Alors, $u^\# \equiv_{\mathfrak{M}} v^\#$, et comme $\equiv_{\mathfrak{M}}$ est compatible avec la déstandardisation, nous avons $\text{ev}(u^\#) = \text{ev}(v^\#)$ et $\text{std}(u^\#) \equiv_{\mathfrak{M}} \text{std}(v^\#)$. Maintenant, d'après le lemme 4.1.10, on a $\text{std}(u)^\# \equiv_{\mathfrak{M}} \text{std}(v)^\#$, et ainsi, $\text{std}(u) \equiv \text{std}(v)$. Réciproquement, si $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv \text{std}(v)$, nous avons $\text{std}(u)^\# \equiv_{\mathfrak{M}} \text{std}(v)^\#$ et $\text{ev}(u^\#) = \text{ev}(v^\#)$. On a, d'après le lemme 4.1.10, $\text{std}(u^\#) \equiv_{\mathfrak{M}} \text{std}(v^\#)$, et, comme M est compatible avec la déstandardisation, $u^\# \equiv_{\mathfrak{M}} v^\#$, ce qui implique finalement $u \equiv v$. \square

4.2 La congruence de Baxter

Nous définissons dans ce paragraphe le monoïde de Baxter et dégageons ses premières propriétés.

4.2.1 Permutations de Baxter et couples d'arbres binaires jumeaux

Rappelons qu'une *permutation de Baxter* [Bax64] est une permutation σ telle que pour tout sous-mot $u := u_1 u_2 u_3 u_4$ de σ où les lettres u_2 et u_3 sont adjacentes dans σ , on a $\text{std}(u) \notin \{2413, 3142\}$. En d'autres termes, σ est une permutation de Baxter si elle évite les *motifs généralisés de permutation* $2-41-3$ et $3-14-2$ (voir [BS00] pour une introduction sur les motifs généralisés de permutation). Par exemple, **42173856** n'est pas une permutation de Baxter puisqu'elle admet le sous-mot 4735 où les lettres 7 et 3 sont adjacentes et $\text{std}(4732) = 2413$. En revanche, 436975128 est une permutation de Baxter. Nous notons \mathfrak{S}^B la classe combinatoire des permutations de Baxter où la taille d'un élément est sa taille en tant que permutation.

Les cardinaux des ensembles \mathfrak{S}_n^B constituent la suite **A001181** de [Slo] qui débute par

$$1, 1, 2, 6, 22, 92, 422, 2074, 10754, 58202, 326240. \quad (4.2.1)$$

Définition 4.2.1. Un couple d'arbres binaires jumeaux est un couple (T_L, T_R) d'arbres binaires tels que la canopée de T_L est complémentaire de la canopée de T_R . La classe combinatoire des couples d'arbres binaires jumeaux est notée \mathcal{J} . La taille d'un couple d'arbres binaires jumeaux (T_L, T_R) est la taille de l'arbre binaire T_L (ou de manière équivalente, la taille de T_R).

La figure 4.1 montre un exemple de couple d'arbres binaires jumeaux.

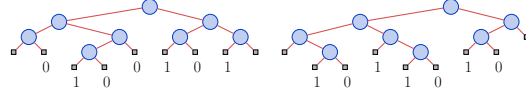


FIGURE 4.1 – Un couple d'arbres binaires jumeaux de taille 8.

Dulucq et Guibert [DG94] ont mis en évidence une bijection entre les permutations de Baxter et les couples d'arbres binaires jumeaux. La bijection que nous obtenons entre ces deux objets est très similaire.

Nous introduisons une notion de *couple d'arbres binaires de recherche jumeaux*, qui est aux couples d'arbres binaires jumeaux ce que la notion d'arbre binaire de recherche est aux arbres binaires :

Définition 4.2.2. Un couple d'arbres binaires de recherche jumeaux (T_L, T_R) est un couple d'arbres binaires jumeaux tel que T_L est un arbre binaire de recherche gauche, T_R est un arbre binaire de recherche droit, et les lectures infixes de T_L et de T_R sont les mêmes.

La figure 4.2 montre un couple d'arbres binaires de recherche jumeaux.

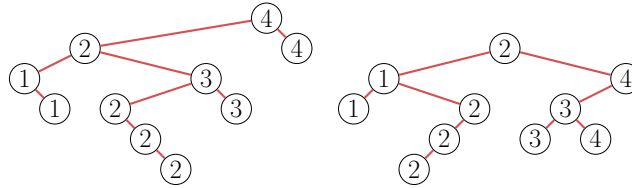


FIGURE 4.2 – Un couple d'arbres binaires de recherche jumeaux de taille 10.

La *forme* d'un couple d'arbres binaires de recherche jumeaux (T_L, T_R) est le couple d'arbres binaires jumeaux (T'_L, T'_R) tel que T'_L est la forme de T_L et T'_R est la forme de T_R .

4.2.2 La relation d'équivalence de Baxter et ses premières propriétés

Définition 4.2.3. Le monoïde de Baxter est le quotient du monoïde libre A^* par la congruence de Baxter \equiv_B obtenue à partir des relations d'adjacence de Baxter $\xrightarrow{B1}$ et $\xrightarrow{B2}$ définies pour $u, v \in A^*$ et $a, b, c, d \in A$ par

$$c u a d v b \xrightarrow{B1} c u d a v b \quad \text{où} \quad a \leq b < c \leq d, \quad (4.2.2)$$

$$b u d a v c \xrightarrow{B2} b u a d v c \quad \text{où} \quad a < b \leq c < d. \quad (4.2.3)$$

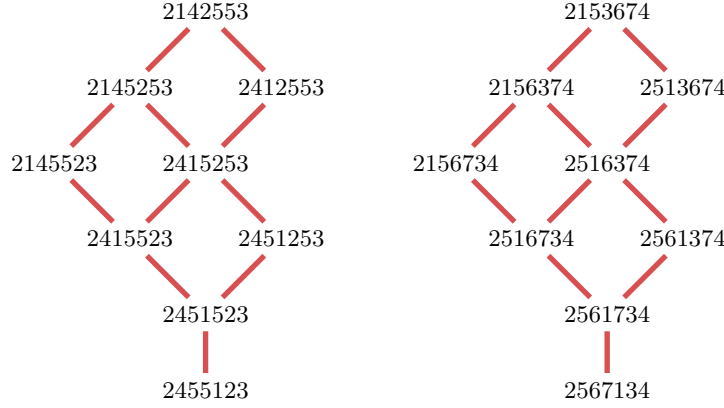


FIGURE 4.3 – La classe d'équivalence de Baxter du mot $u := 2415253$ et de la permutation $2516374 = \text{std}(u)$. Les arêtes représentent les relations d'adjacence de Baxter.

Par exemple, la classe d'équivalence de Baxter de 2415253 (voir la figure 4.3) est

$$\{2142553, 2145253, 2145523, 2412553, 2415253, 2415523, 2451253, 2451523, 2455123\}. \quad (4.2.4)$$

Notons que si l'on restreint la congruence de Baxter aux mots sans répétition de lettres, les relations d'adjacence $\xleftrightarrow{B1}$ et $\xleftrightarrow{B2}$ peuvent alors être remplacées par l'unique relation d'adjacence \xleftrightarrow{B} qui vérifie

$$x u a d v y \xleftrightarrow{B} x u d a v y \quad \text{où} \quad a < x, y < d. \quad (4.2.5)$$

Proposition 4.2.4. *Le monoïde de Baxter est compatible avec la déstandardisation.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. Soient $u, v \in A^*$ tels que $u \xleftrightarrow{B1} v$. Nous avons alors $u = xcyadzbt$ et $v = xcydazbt$ avec $x, y, z, t \in A^*$ et $a \leq b < c \leq d \in A$. Comme $\xleftrightarrow{B1}$ agit en permutant des lettres, on a $\text{ev}(u) = \text{ev}(v)$. De plus, les lettres a', b', c' et d' de $\text{std}(u)$ respectivement aux mêmes positions que les lettres a, b, c et d dans u vérifient $a' < b' < c' < d'$ à cause de leurs positions relatives dans $\text{std}(u)$ et les relations d'ordre entre a, b, c et d . Les mêmes relations existent pour les lettres de $\text{std}(v)$, montrant que $\text{std}(u) \xleftrightarrow{B1} \text{std}(v)$. La démonstration pour le cas $u \xleftrightarrow{B2} v$ est analogue.

Réciproquement, supposons que v est une permutation de u et $\text{std}(u) \xleftrightarrow{B1} \text{std}(v)$. On a $\text{std}(u) = xcyadzbt$ et $\text{std}(v) = xcydazbt$ avec $x, y, z, t \in A^*$ et $a < b < c < d \in A$. Le mot u est une version non standardisée de $\text{std}(u)$ de sorte que les lettres a', b', c' et d' de u respectivement aux mêmes positions que les lettres a, b, c et d de $\text{std}(u)$ vérifient $a' \leq b' < c' \leq d'$ à cause de leurs positions relatives dans u et les relations d'ordre entre a, b, c et d . Les mêmes relations existent pour les lettres de v , montrant que $u \xleftrightarrow{B1} v$. La démonstration pour le cas $u \xleftrightarrow{B2} v$ est analogue. \square

Proposition 4.2.5. *Le monoïde de Baxter est compatible aux restrictions aux intervalles d'alphabet.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $a < x, y < d$. Soit I un intervalle de $\{1, \dots, n\}$ et $R := I \cap \{a, x, y, d\}$. Si $R = \{a, x, y, d\}$, alors

$$\sigma|_I = t|_I x u|_I a d v|_I y w|_I \quad \text{et} \quad \nu|_I = t|_I x u|_I d a v|_I y w|_I, \quad (4.2.6)$$

de sorte que $\sigma|_I \xleftrightarrow{B} \nu|_I$. Autrement, on a $\sigma|_I = \nu|_I$ et ainsi, $\sigma|_I \equiv_B \nu|_I$. \square

Proposition 4.2.6. *Le monoïde de Baxter est compatible avec l'involution de Schützenberger.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $a < x, y < d$. Nous avons

$$\sigma^\# = w^\# y^\# v^\# d^\# a^\# u^\# x^\# t^\# \quad \text{et} \quad \nu^\# = w^\# y^\# v^\# a^\# d^\# u^\# x^\# t^\#. \quad (4.2.7)$$

Étant donné que $d^\# < x^\#, y^\# < a^\#$, nous avons $\sigma^\# \xleftrightarrow{B} \nu^\#$. \square

4.2.3 Liens avec d'autres bons monoïdes connus

Nous mettons ici en évidence un lien entre le monoïde de Baxter et les monoïdes sylvestre et $\#$ -sylvestre. Ce lien, riche en conséquences pour le reste du chapitre, permet de montrer que le monoïde de Baxter s'obtient à partir du monoïde sylvestre en l'intersectant avec son image par l'involution de Schützenberger. L'une des premières applications de cette propriété apparaît dans la démonstration du fait que les classes d'équivalence de Baxter de permutations forment des intervalles du permutoèdre. Nous donnons aussi un lien entre le monoïde de Baxter et le monoïde des 3-reculs.

Lien avec les monoïdes sylvestre et $\#$ -sylvestre

Montrons pour commencer une propriété sur les classes sylvestres de permutations qui nous sera utile par la suite.

Lemme 4.2.7. *Soient $u := xac y$ et $v := xca y$ deux mots tels que $x, y \in A^*$, $a < c \in A$ et $u \equiv_s v$. Alors, $u \xleftrightarrow{S} v$.*

Démonstration. Ce lemme est une conséquence du théorème 4.1.6. Comme u et v donnent le même arbre binaire de recherche droit T en les insérant de droite à gauche, le nœud étiqueté par a et le nœud étiqueté par c dans T ne peuvent pas être ancêtres l'un de l'autre. Ceci implique qu'il existe un nœud étiqueté par une lettre b , ancêtre commun aux nœuds étiquetés par a et par c tel que $a \leq b < c$. Ainsi, $u \xleftrightarrow{S} v$. \square

Le lemme 4.2.7 montre également que les relations d'adjacence \xleftrightarrow{S} de toute classe sylvestre C de permutations sont exactement les relations de couverture du permutoèdre restreint aux éléments de C . Notons qu'une démonstration analogue du lemme 4.2.7 montre que c'est aussi le cas pour la relation d'adjacence $\xleftrightarrow{\#S}$ du monoïde $\#$ -sylvestre.

Le monoïde de Baxter, le monoïde sylvestre, et le monoïde $\#$ -sylvestre sont liés de la manière suivante :

Proposition 4.2.8. *Soient $u, v \in A^*$. Alors, $u \equiv_B v$ si et seulement si $u \equiv_s v$ et $u \equiv_{\#s} v$.*

Démonstration. Une fois de plus, il suffit de montrer la propriété pour les relations d'adjacence de Baxter. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = xxyadz y t$ et $\nu = xxydzay t$ avec $a < x, y < d$. La présence des lettres a, d et y avec $a < y < d$ assure $\sigma \xleftrightarrow{S} \nu$. En outre, la présence des lettres x, a et d avec $a < x < d$ assure $\sigma \xleftrightarrow{\#S} \nu$.

Réciproquement, étant donné que les monoïdes sylvestre et $\#$ -sylvestre sont compatibles avec la déstandardisation [HNT05], il suffit de montrer la propriété pour les permutations.

Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \equiv_s \nu$ et $\sigma \equiv_{\#s} \nu$. Posons $\tau := \inf_{\leq_P} \{\sigma, \nu\}$. Comme le permutoèdre est un treillis, τ est bien défini, et comme les classes d'équivalence sylvestres et $\#$ -sylvestres de permutations sont des intervalles du permutoèdre [HNT05], nous avons $\sigma \equiv_s \tau \equiv_s \nu$ et $\sigma \equiv_{\#s} \tau \equiv_{\#s} \nu$. De plus, par le lemme 4.2.7, et encore par le fait que les classes d'équivalence sylvestres et $\#$ -sylvestres de permutations sont des intervalles du permutoèdre, pour toute chaîne saturée $\tau \leq_P \pi \leq_P \dots \leq_P \sigma$ et $\tau \leq_P \mu \leq_P \dots \leq_P \nu$, il existe les suites de relations d'adjacence $\tau \xrightarrow{s} \pi \xrightarrow{s} \dots \xrightarrow{s} \sigma$, $\tau \xrightarrow{\#s} \pi \xrightarrow{\#s} \dots \xrightarrow{\#s} \sigma$, $\tau \xrightarrow{s} \mu \xrightarrow{s} \dots \xrightarrow{s} \nu$ et $\tau \xrightarrow{\#s} \mu \xrightarrow{\#s} \dots \xrightarrow{\#s} \nu$. Par conséquent, $\tau \equiv_B \sigma$ et $\tau \equiv_B \nu$, ce qui implique $\sigma \equiv_B \nu$. \square

La proposition 4.2.8 montre que les classes d'équivalence de Baxter sont des intersections de classes d'équivalences sylvestre et $\#$ -sylvestres.

À partir de la caractérisation des classes d'équivalence de Baxter fournie par la proposition 4.2.8 et en restreignant la relation \equiv_B aux permutations, on obtient :

Proposition 4.2.9. *Pour tout $n \geq 0$, toute classe d'équivalence de $\mathfrak{S}_n / \equiv_B$ est un intervalle du permutoèdre.*

Démonstration. Par la proposition 4.2.8, les classes de Baxter sont des intersections de classes sylvestres et $\#$ -sylvestres. De plus, les classes d'équivalence de permutations sous les relations sylvestres et $\#$ -sylvestres sont des intervalles du permutoèdre [HNT05]. La proposition provient du fait que l'intersection de deux intervalles dans un treillis est également un intervalle. \square

Lemme 4.2.10. *Soient $u := x \mathbf{a} \mathbf{d} \mathbf{y}$ et $v := x \mathbf{d} \mathbf{a} \mathbf{y}$ tels que $x, y \in A^*$, $\mathbf{a} < \mathbf{d} \in A$ et $u \equiv_B v$. Alors, $u \xrightarrow{B1} v$ ou $u \xrightarrow{B2} v$.*

Démonstration. D'après la proposition 4.2.8, étant donné que $u \equiv_B v$, nous avons $u \equiv_s v$ et ainsi, par le lemme 4.2.7, $u \xrightarrow{s} v$. Ceci implique l'existence d'une lettre \mathbf{y} dans le facteur \mathbf{y} vérifiant $\mathbf{a} \leq \mathbf{y} < \mathbf{d}$. De la même manière, nous avons aussi $u \equiv_{\#s} v$ et ainsi $u \xrightarrow{\#s} v$. Ceci implique l'existence d'une lettre \mathbf{x} dans le facteur \mathbf{x} vérifiant $\mathbf{a} < \mathbf{x} \leq \mathbf{d}$. Par conséquent, u et v sont adjacents par $\xrightarrow{B1}$ ou $\xrightarrow{B2}$. \square

Le lemme 4.2.10 est l'analogue, dans le cas de la relation d'équivalence de Baxter, du lemme 4.2.7 et montre aussi que les relations d'adjacence de Baxter de toute classe de Baxter C de permutations sont exactement les relations de couverture du permutoèdre restreint aux éléments de C .

Lien avec le monoïde des 3-reculs

Le monoïde de Baxter et le monoïde des 3-reculs sont liés de la manière suivante :

Proposition 4.2.11. *Toute classe d'équivalence de \equiv_{3R} peut être exprimée comme une union de classes de Baxter.*

Démonstration. Ceci revient à montrer que pour tous mots u et v , $u \equiv_B v$ implique $u \equiv_{3R} v$. D'après la proposition 4.2.4, il suffit de montrer cette propriété sur les permutations. Supposons ainsi $\sigma \equiv_B \nu$ où σ et ν sont des permutations. Il suffit maintenant de montrer la propriété pour les relations d'adjacence de Baxter. Supposons par conséquent que $\sigma \xrightarrow{B1} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $\mathbf{a} < \mathbf{x}, \mathbf{y} < \mathbf{d}$. Comme σ et ν sont des permutations, $\mathbf{x} \neq \mathbf{t}$ et ainsi, $\mathbf{a} < \mathbf{x} < \mathbf{y} < \mathbf{d}$ ou $\mathbf{a} < \mathbf{y} < \mathbf{x} < \mathbf{d}$. Il existe donc au minimum les deux lettres \mathbf{x} et \mathbf{y} qui se situent en valeur entre \mathbf{a} et \mathbf{d} dans σ et dans ν . On en conclut ainsi que $\sigma \equiv_{3R} \nu$. \square

Observons que la proposition 4.2.11 n'est plus vraie pour la relation \equiv_{4R} puisqu'il y a 22 classes d'équivalence de permutations de taille 4 pour \equiv_B mais 24 pour \equiv_{4R} . Réciproquement, la relation \equiv_{4R} n'est pas non plus un raffinement de \equiv_B puisque pour tout $n \geq 5$, la permutation $1 \cdot n \cdot n-1 \dots 2$ est la seule de sa classe de Baxter mais pas de sa classe pour \equiv_{4R} .

De plus, étant donné que les classes d'équivalence de \equiv_{kR} sont des unions de classes de \equiv_{k+1R} , d'après la proposition 4.2.11, les classes hypoplaxiques sont des unions de classes de Baxter. Le monoïde de Baxter peut donc être vu comme une généralisation du monoïde hypoplaxique.

4.3 Une correspondance à la Robinson-Schensted

Nous décrivons un algorithme permettant de décider de l'équivalence de deux mots conformément à la congruence de Baxter. Étant donné un mot $u \in A^*$, cet algorithme produit son \mathbb{P} -symbole, un couple d'arbres binaires de recherche jumeaux (T_L, T_R) (voir la définition 4.2.2) tel que les lectures infixes de T_L et de T_R donnent toutes deux le réarrangement croissant de u . Il produit aussi son \mathbb{Q} -symbole, un couple d'arbres binaires jumeaux (S_L, S_R) tel que S_L (resp. S_R) est un arbre binaire croissant (resp. décroissant), et les lectures infixes de S_L et de S_R sont les mêmes.

La principale motivation d'introduire cet algorithme, en plus du fait qu'il fournit un analogue de l'algorithme classique d'insertion dans un tableau de Young, est qu'il permet de montrer que les classes de Baxter de permutations de taille n sont en bijection avec les couples d'arbres binaires jumeaux non étiquetés, et donc, par les résultats de Dulucq et Guibert [DG94], également en bijection avec les permutations de Baxter de taille n . L'autre motivation est de fournir un analogue de la correspondance de Robinson-Schensted en montrant que l'application $u \mapsto (\mathbb{P}(u), \mathbb{Q}(u))$ est une bijection entre les mots de A^* et l'ensemble des couples $((T_L, T_R), (S_L, S_R))$ décrits ci-dessus.

Nous allons commencer par établir le fait que pour tout mot u , le \mathbb{P} -symbole de u respecte les conditions que nous venons d'énoncer. Ensuite, nous montrons que l'on a bien $u \equiv_B v$ si et seulement si $\mathbb{P}(u) = \mathbb{P}(v)$, ce qui établira une injection entre l'ensemble des classes de Baxter de permutations de taille n et l'ensemble des couples d'arbres binaires jumeaux de taille n . La surjection entre ces deux ensembles va suivre d'un algorithme d'extraction qui, étant donné un couple d'arbres binaires jumeaux, calcule l'unique permutation de Baxter de sa classe. Nous donnons de plus un algorithme qui extrait la plus petite permutation de sa classe, ce qui mène à un nouvel algorithme pour calculer la bijection entre les permutations de Baxter et les *permutations de Baxter tordues* de Reading [Rea04], [LR10].

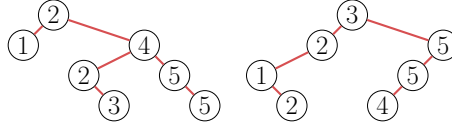
Nous terminons cette partie en donnant une version itérative de l'algorithme du \mathbb{P} -symbole, qui a pour mérite de permettre de calculer dans le monoïde de Baxter.

4.3.1 Un algorithme d'insertion à la Robinson-Schensted

Définition 4.3.1. *Le \mathbb{P} -symbole de Baxter — ou simplement \mathbb{P} -symbole lorsque le contexte est clair — d'un mot $u \in A^*$ est le couple (T_L, T_R) où T_L (resp. T_R) est l'arbre binaire de recherche gauche (resp. droit) obtenu en insérant en feuille les lettres de u de gauche à droite (resp. droite à gauche).*

À titre d'exemple, la figure 4.4 donne le \mathbb{P} -symbole du mot 2415253.

Avant de donner la démonstration du fait que ce \mathbb{P} -symbole respecte les conditions que nous avons énoncées dans l'introduction de cette partie, donnons une explication intuitive de sa validité.

FIGURE 4.4 – Le \mathbb{P} -symbole de Baxter du mot 2415253.

Soit u un mot. D'après le théorème 4.1.6, l'arbre binaire de recherche gauche T_L obtenu par des insertions en feuille des lettres de u de la gauche vers la droite encode sa classe $\#$ -sylvestre, et que, de la même manière, l'arbre binaire de recherche droit T_R obtenu par des insertions en feuille des lettres de u de la droite vers la gauche encode sa classe sylvestre. Tel que nous l'avons défini, le \mathbb{P} -symbole de u encode à la fois sa classe sylvestre et sa classe $\#$ -sylvestre. De plus, en se basant sur la proposition 4.2.8, on observe que deux mots u et v vérifient $u \equiv_{\mathbb{B}} v$ si et seulement si $u \equiv_{\mathbb{S}} v$ et $u \equiv_{\# \mathbb{S}} v$. Ainsi, l'information contenue dans le \mathbb{P} -symbole d'un mot encode exactement sa classe de Baxter.

Soit T un arbre binaire. Nous appelons i^{e} nœud de T le i^{e} nœud parcouru lors du parcours infixe de T . De même, nous appelons i^{e} feuille de T la feuille en position i lorsque l'on considère les feuilles de T de la gauche vers la droite.

La validité du \mathbb{P} -symbole de Baxter repose sur les deux lemmes et la proposition suivante.

Lemme 4.3.2. *Soit T un arbre binaire non vide et x sa i^{e} feuille. Si x est orientée vers la gauche, alors x est attachée au i^{e} nœud interne de T . Si x est orientée vers la droite, alors x est attachée au $(i-1)^{\text{e}}$ nœud de T .*

Démonstration. Nous procédons par induction structurelle sur l'ensemble des arbres binaires non vides. Si T est l'arbre binaire à un nœud, le lemme est vérifié. Autrement, on a $T = A \wedge B$. Soit x la i^{e} feuille de T et y le nœud auquel x est attachée. Si x se trouve aussi dans A et $A = \perp$, alors x est orientée vers la gauche et est attachée à la racine de T , et le lemme est vérifié car la racine de T est dans ce cas le premier nœud de T . Si x est dans A et $A \neq \perp$, x est également la i^{e} feuille de A et y est dans A , de sorte que le lemme s'ensuit par hypothèse d'induction sur A . Sinon, x est dans B . Si $B = \perp$, x est orientée vers la droite et est attachée à la racine de T , et le lemme est vérifié car la racine de T est dans ce cas le dernier nœud de T . Autrement, x est la $(i-n-1)^{\text{e}}$ feuille de B où n est le nombre de nœuds de A . Supposons que y est le j^{e} nœud de T . Alors, y est le $(j-n-1)^{\text{e}}$ nœud de B . Par conséquent, le lemme s'ensuit par hypothèse d'induction sur B . \square

Lemme 4.3.3. *Soit σ une permutation et T l'arbre binaire de recherche obtenu en insérant les lettres de σ de la gauche vers la droite. Alors, la $(i+1)^{\text{e}}$ feuille de T est orientée vers la droite si et seulement si i est un recul de σ .*

Démonstration. Posons $\mathbf{a} := i$ et $\mathbf{b} := i+1$. Supposons que \mathbf{a} est un recul de σ . Dans ce cas, σ est de la forme $\sigma = ubvaw$. Étant donné qu'aucune lettre \mathbf{x} des mots u et v ne vérifie $\mathbf{a} < \mathbf{x} < \mathbf{b}$, le nœud de T étiqueté par \mathbf{b} possède un nœud étiqueté par \mathbf{a} dans son sous-arbre gauche, lui-même n'ayant pas de sous-arbre droit et ainsi contribue, par le lemme 4.3.2, à une feuille orientée vers la droite en position $i+1$.

Réciproquement, supposons que \mathbf{a} n'est pas un recul de σ . Dans ce cas, σ est de la forme $\sigma = uavbw$. Pour la même raison que dans le cas précédent, le nœud de T étiqueté par \mathbf{a} possède un nœud étiqueté par \mathbf{b} dans son sous-arbre droit, lui-même n'ayant pas de sous-arbre gauche, et ainsi contribue, par le lemme 4.3.2, à une feuille orientée vers la gauche en position $i+1$. \square

Proposition 4.3.4. *Pour tout mot $u \in A^*$, le \mathbb{P} -symbole (T_L, T_R) de u est un couple d'arbres binaires de recherche jumeaux tel que les lectures infixes de T_L et T_R donnent toutes deux le réarrangement croissant de u .*

Démonstration. Notons que par définition de l'algorithme INSERTIONFEUILLE, T_L (resp. T_R) est un arbre binaire de recherche gauche (resp. droit), et que les lectures infixes de T_L et T_R donnent toutes deux le réarrangement croissant de u . Étant donné que les insertions en feuille des lettres de u et de $\text{std}(u)$ de la gauche vers la droite (resp. de la droite vers la gauche) dans un arbre binaire de recherche gauche (resp. droit) donnent des arbres binaires de même forme, nous pouvons considérer que $u =: \sigma$ est une permutation. Ainsi, le lemme 4.3.3 implique que les canopées de T_L et de T_R sont complémentaires puisque i est un recul de σ si et seulement si i n'est pas un recul de σ^\sim . Ceci montre que (T_L, T_R) est un couple d'arbres binaires jumeaux. \square

Théorème 4.3.5. *Soient $u, v \in A^*$. Alors, $u \equiv_{\mathbb{B}} v$ si et seulement si $\mathbb{P}(u) = \mathbb{P}(v)$.*

Démonstration. Supposons $u \equiv_{\mathbb{B}} v$. Par la proposition 4.2.8, u et v sont équivalents pour les congruences sylvestre et $\#$ -sylvestre. Par conséquent, d'après le théorème 4.1.6, u et v ont le même \mathbb{P} -symbole sylvestre et $\#$ -sylvestre, et ainsi, $\mathbb{P}(u) = \mathbb{P}(v)$.

Réciproquement, supposons $\mathbb{P}(u) = \mathbb{P}(v) =: (T_L, T_R)$. Comme les insertions en feuille de u et de v de la gauche vers la droite donnent toutes deux T_L , nous avons, par le théorème 4.1.6, $u \equiv_{\#s} v$. En outre, les insertions en feuille de u et de v de la droite vers la gauche donnent toutes deux T_R , de sorte que, par le théorème 4.1.6, $u \equiv_s v$. Ainsi, d'après la proposition 4.2.8, $u \equiv_{\mathbb{B}} v$. \square

Remarquons qu'une classe d'équivalence de Baxter de permutations peut être encodée par un couple non étiqueté d'arbres binaires jumeaux. Il existe en effet un unique moyen d'étiqueter bijectivement un arbre binaire à n nœuds sur l'ensemble $[n]$ de sorte qu'il soit un arbre binaire de recherche, et à plus forte raison, il existe un unique moyen d'étiqueter ainsi chacun des arbres d'un couple d'arbres binaires jumeaux. Par conséquent, dans ce qui suit, les couples non étiquetés d'arbres binaires jumeaux seront vus comme des couples d'arbres binaires jumeaux étiquetés par une permutation.

4.3.2 Une correspondance à la Robinson-Schensted

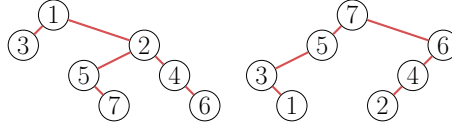
Avant de définir un \mathbb{Q} -symbole nécessaire pour établir un analogue de la correspondance de Robinson-Schensted, rappelons deux constructions classiques qui associent à un mot u un arbre binaire. L'arbre binaire croissant $\text{croi}(u)$ de u est défini récursivement par

$$\text{croi}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{croi}(v) \wedge_{\mathbf{a}} \text{croi}(w) & \text{où } u = v\mathbf{a}w, \mathbf{a} = \min \text{Alph}(u), \\ & \text{et } \mathbf{a} < \mathbf{b} \text{ pour tout } \mathbf{b} \in \text{Alph}(v). \end{cases} \quad (4.3.1)$$

Quant à lui, l'arbre binaire décroissant $\text{decr}(u)$ de u est défini récursivement par

$$\text{decr}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{decr}(v) \wedge_{\mathbf{b}} \text{decr}(w) & \text{où } u = v\mathbf{b}w, \mathbf{b} = \max \text{Alph}(u), \\ & \text{et } \mathbf{a} < \mathbf{b} \text{ pour tout } \mathbf{b} \in \text{Alph}(w). \end{cases} \quad (4.3.2)$$

Définition 4.3.6. *Le \mathbb{Q} -symbole de Baxter — ou simplement \mathbb{Q} -symbole lorsque le contexte est clair — d'un mot $u \in A^*$ est le couple (S_L, S_R) où $S_L := \text{croi}(\text{std}(u)^{-1})$ et $S_R := \text{decr}(\text{std}(u)^{-1})$.*

FIGURE 4.5 – Le \mathbb{Q} -symbole de Baxter du mot 2415253.

À titre d'exemple, la figure 4.5 représente le \mathbb{Q} -symbole du mot 2415253, dont l'inverse du standardisé est 3157246.

Il est immédiat, étant donné un mot u , que le \mathbb{Q} -symbole de u permet, en association avec son \mathbb{P} -symbole, de reconstruire u . En effet, si $\mathbb{P}(u) = (T_L, T_R)$ et $\mathbb{Q}(u) = (S_L, S_R)$, le couple (T_R, S_R) constitue à lui seul les \mathbb{P} et \mathbb{Q} -symboles sylvestres de u [HNT05], et ce couple suffit pour reconstruire u . Pour calculer le mot correspondant au couple $((T_L, T_R), (S_L, S_R))$, il suffit de lire les étiquettes de T_R dans l'ordre imposé par les étiquettes des nœuds correspondants de S_R . La même propriété est vraie pour T_L et S_L .

Pour montrer l'analogie de la correspondance de Robinson-Schensted pour le monoïde de Baxter, nous avons besoin du lemme et de la proposition suivante. Le lemme est tiré de [HNT05], reformulé dans notre contexte, et complété dans le cas des arbres binaires croissants.

Lemme 4.3.7. *Soit u un mot et $\sigma := \text{std}(u)^{-1}$. L'arbre binaire de recherche droit (resp. gauche) obtenu en insérant les lettres de u de la droite vers la gauche (resp. de la gauche vers la droite) et $\text{decr}(\sigma)$ (resp. $\text{croi}(\sigma)$) ont la même forme.*

Proposition 4.3.8. *Pour tout mot $u \in A^*$, la forme du \mathbb{Q} -symbole (S_L, S_R) de u est un couple d'arbres binaires jumeaux. De plus, S_L est un arbre binaire croissant, S_R un arbre binaire décroissant, et leur lecture infixe est la même et égale à $\text{std}(u)^{-1}$.*

Démonstration. Par définition du \mathbb{Q} -symbole, S_L et S_R sont respectivement les arbres binaires croissants et décroissants de $\sigma := \text{std}(u)^{-1}$. D'après le lemme 4.3.7, un arbre binaire de la même forme que S_L (resp. S_R) peut être obtenu en insérant en feuille les lettres de σ^{-1} de gauche à droite (resp. de droite à gauche). Ainsi, par le lemme 4.3.3, la forme de (S_L, S_R) est un couple d'arbres binaires jumeaux. De plus, par définition des constructions croi et decr , il suit par récurrence sur la taille de σ que S_L et S_R admettent bien σ comme lecture infixe. \square

Théorème 4.3.9. *L'application $u \mapsto (\mathbb{P}(u), \mathbb{Q}(u))$ définit une bijection entre les éléments de A^* et l'ensemble des couples $((T_L, T_R), (S_L, S_R))$ où*

- (i) (T_L, T_R) est un couple d'arbres binaires de recherche jumeaux;
- (ii) (S_L, S_R) est un couple d'arbres binaires jumeaux tel que S_L (resp. S_R) est un arbre binaire croissant (resp. décroissant), et S_L et S_R possèdent la même lecture infixe;
- (iii) (T_L, T_R) et (S_L, S_R) sont de même forme.

Démonstration. Commençons par montrer que pour tout $u \in A^*$, le couple $(\mathbb{P}(u), \mathbb{Q}(u))$ vérifie les trois points du théorème. Il suit de la proposition 4.3.4 que (i) est vérifié, et de la proposition 4.3.8 que (ii) l'est aussi. De plus, par le lemme 4.3.7, (iii) est établi. En outre, comme nous l'avons mentionné et justifié précédemment, il est possible de reconstruire de manière unique u à partir du couple $(\mathbb{P}(u), \mathbb{Q}(u))$. Ceci montre que la correspondance est bien définie et qu'elle est injective.

Réciproquement, supposons que $((T_L, T_R), (S_L, S_R))$ vérifie les trois points du théorème. D'après [HNT02], il existe une bijection entre les éléments de A^* et les couples (T_R, S_R) où T_R est un arbre binaire de recherche droit et S_R un arbre binaire décroissant de la même forme.

Soit u le mot en bijection avec (T_R, S_R) . De la même manière, il existe une bijection entre les éléments de A^* et les couples (T_L, S_L) où T_L est un arbre binaire de recherche gauche et S_L un arbre binaire croissant de la même forme. Soit v le mot en bijection avec (T_L, S_L) . Par hypothèse, T_L et T_R ont tous les deux la même lecture infixe, ce qui implique $\text{ev}(u) = \text{ev}(v)$. De la même manière, comme S_L et S_R ont tous les deux la même lecture infixe, $\text{std}(u)^{-1} = \text{std}(v)^{-1}$ et par conséquent, $u = v$. Observons de plus que le couple (T_L, S_L) est entièrement déterminé par le couple (T_R, S_R) et inversement. Maintenant, toujours d'après [HNT02], le couple (T_R, S_R) est le \mathbb{P} -symbole sylvestre de u , et le couple (T_L, S_L) est \mathbb{P} -symbole $\#$ -sylvestre de u . Ainsi, l'insertion de u donne le couple $((T_L, T_R), (S_L, S_R))$, montrant que l'application du théorème est également surjective. \square

4.3.3 Extraction de mots

Avant de présenter trois algorithmes qui permettent d'extraire des permutations et des mots particuliers à partir d'un couple d'arbres binaires de recherche jumeaux, commençons par une remarque sur les classes d'équivalence de Baxter qui nous sera utile par la suite.

Les classes de Baxter en tant qu'extensions linéaires

Soit T un arbre binaire étiqueté à n nœuds. Notons par $\Delta(T)$ (resp. $\nabla(T)$) le poset (N, \leq) où N est l'ensemble des nœuds de T , et \leq est défini pour $x, y \in N$ par

$$x \leq y \quad \text{si} \quad x \text{ est un ancêtre (resp. un descendant) de } y. \quad (4.3.3)$$

Si la séquence x_1, \dots, x_n est une extension linéaire de $\Delta(T)$ (resp. $\nabla(T)$), nous dirons également par abus de langage que le mot $u_1 \dots u_n$ est une *extension linéaire* de $\Delta(T)$ (resp. $\nabla(T)$) si pour tout $i \in [n]$, l'étiquette du nœud x_i est u_i .

La Note 4 de [HNT05] nous renseigne sur le fait que les mots d'une classe sylvestre encodée par un arbre binaire de recherche droit T_R sont exactement les extensions linéaires de $\nabla(T_R)$. Cette propriété est adaptable dans le contexte des classes $\#$ -sylvestres, et implique que les mots d'une classe $\#$ -sylvestre encodée par un arbre binaire de recherche gauche T_L sont exactement les extensions linéaires de $\Delta(T_L)$. De plus, d'après la proposition 4.2.8, les mots d'une classe de Baxter figurent à la fois dans une même classe sylvestre et une même classe $\#$ -sylvestre. Ainsi, les mots d'une classe de Baxter encodée par un couple d'arbres binaires de recherche jumeaux (T_L, T_R) sont exactement les mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$.

Nous notons $\Delta(T_L) \cup \nabla(T_R)$ le poset qui regroupe les relations d'ordre de $\Delta(T_L)$ et de $\nabla(T_R)$ de la manière suivante. On considère comme égaux deux nœuds x de T_L et x' de T_R s'ils se trouvent en même position dans leurs arbres binaires respectifs. Le poset $\Delta(T_L) \cup \nabla(T_R)$ est défini sur l'ensemble des nœuds de T_L (ou de manière équivalente avec l'identification précédente, sur les nœuds de T_R) par la relation d'ordre \leq définie par

$$x \leq y \quad \text{si} \quad x \leq y \text{ dans } \Delta(T_L) \quad \text{ou} \quad x \leq y \text{ dans } \nabla(T_R). \quad (4.3.4)$$

Remarquons qu'il s'agit bien là d'un poset, *i.e.*, la relation d'ordre définie en (4.3.4) est bien antisymétrique. En effet, si l'on suppose par l'absurde que l'on a $x \leq y$ dans $\Delta(T_L)$ et $y \leq x$ dans $\nabla(T_R)$, cela signifie que x est un ancêtre de y à la fois dans T_L et dans T_R . Ceci ne peut jamais arriver car, d'après la définition 4.3.1, le \mathbb{P} -symbole (T_L, T_R) est défini de telle sorte que T_L soit obtenu par une insertion en feuille de gauche à droite d'un mot, et T_R , de droite à gauche de ce même mot.

On observe que les mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$ sont également des extensions linéaires de $\Delta(T_L) \cup \nabla(T_R)$, et réciproquement. Ainsi, les mots

qui figurent dans la classe de Baxter encodée par (T_L, T_R) sont les extensions linéaires du poset $\Delta(T_L) \cup \nabla(T_R)$.

Par exemple, pour le couple d'arbres binaires de recherche jumeaux suivant :

$$(T_L, T_R) := \left(\begin{array}{c} \text{Arbre } T_L: \text{ racine 2, fils 1 (gauche), 5 (droite). 5 a des fils 4 (gauche) et 7 (droite). 4 a un fils 3 (gauche).} \\ \text{Arbre } T_R: \text{ racine 6, fils 3 (gauche) et 7 (droite). 3 a des fils 1 (gauche) et 4 (droite). 4 a un fils 5 (gauche).} \end{array} \right), \quad (4.3.5)$$

l'ensemble des mots qui sont dans la classe de Baxter encodée par (T_L, T_R) est

$$\{5214376, 5214736, 5217436, 5241376, 5241736, 5247136, 5271436, 5274136, 5721436, 5724136\}, \quad (4.3.6)$$

qui est exactement l'ensemble des mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$.

Pour construire le diagramme de Hasse du poset $\Delta(T_L) \cup \nabla(T_R)$, il suffit de partir de T_L et de lui ajouter les arcs encodant les relations d'ordre induites par T_R qui n'existent pas déjà dans celles induites par T_L .

Ainsi, pour l'exemple précédent, nous obtenons

$$\Delta(T_L) \cup \nabla(T_R) = \begin{array}{c} \text{Diagramme de Hasse:} \\ \text{Nœuds: 1, 2, 3, 4, 5, 6, 7.} \\ \text{Arcs: } 1 \rightarrow 2, 2 \rightarrow 5, 5 \rightarrow 4, 5 \rightarrow 7, 4 \rightarrow 3, 3 \rightarrow 6, 6 \rightarrow 7. \end{array}, \quad (4.3.7)$$

et l'ensemble (4.3.6) contient exactement les extensions linéaires de ce poset.

Extraction de permutations de Baxter

L'algorithme suivant permet de calculer un mot qui appartient à la classe de Baxter encodée par un couple d'arbres binaires de recherche jumeaux (T_L, T_R) en entrée. Lorsque (T_L, T_R) est étiqueté par une permutation, notre algorithme coïncide avec l'algorithme que Dulucq et Guibert ont utilisé pour présenter leur bijection entre les couples d'arbres binaires jumeaux non étiquetés et les permutations de Baxter [DG94]. En outre, étant donné que leur algorithme retourne toujours une permutation de Baxter, notre algorithme retourne une permutation de Baxter sur l'entrée d'un couple d'arbres binaires de recherche jumeaux étiqueté par une permutation.

Algorithme : EXTRAIREBAXTER.

Entrée : un couple d'arbres binaires de recherche jumeaux (T_L, T_R) .

Sortie : un mot figurant dans la classe de Baxter encodée par (T_L, T_R) .

1. Soit $u := \epsilon$ le mot vide.
2. Tant que $T_L \neq \perp$ et $T_R \neq \perp$:
 - (a) Soit a l'étiquette de la racine de T_L .
 - (b) Soit i l'indice de la racine de T_L .
 - (c) Poser $u := u \cdot a$.
 - (d) Soit A (resp. B) le sous-arbre gauche (resp. droit) de T_L .
 - (e) Si le i^{e} nœud de T_R est un fils gauche dans T_R :
 - i. alors, poser $T_L := A / B$.
 - ii. Sinon, poser $T_L := A \setminus B$.

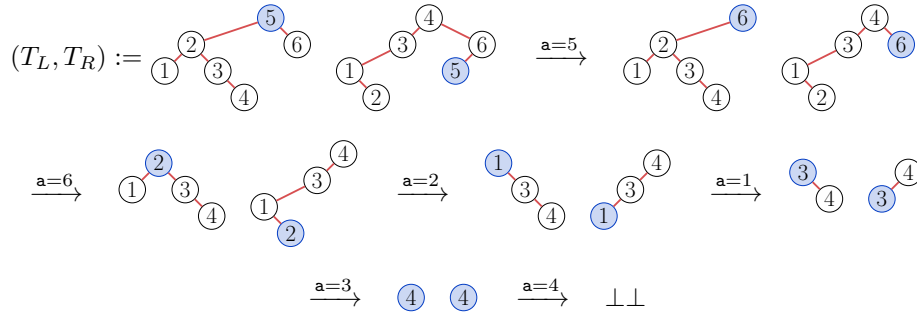


FIGURE 4.6 – Une exécution de l’algorithme EXTRAIREBAXTER sur le couple d’arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation de Baxter calculée est 562134.

- (f) Supprimer le i^{e} nœud de T_R .
3. Retourner u .

Fin.

La figure 4.6 montre une exécution de cet algorithme.

Les résultats de Dulucq et Guibert permettent immédiatement de montrer que l’exécution de EXTRAIREBAXTER se termine pour toute entrée. Il reste seulement à montrer que le mot calculé appartient bien à la classe de Baxter encodée par le couple d’arbres binaires de recherche jumeaux en entrée. Pour cela, nous avons besoin du lemme suivant.

Lemme 4.3.10. *Soit (T_L, T_R) un couple d’arbres binaires jumeaux non vide. Si la racine de T_L est le i^{e} nœud de T_L , alors le i^{e} nœud de T_R n’a pas de fils.*

Démonstration. Supposons que $T_L = A \wedge B$. Notons que si A et B sont vides, T_L et T_R sont réduits à un nœud, et le lemme est vérifié.

Si $A \neq \perp$, supposons par l’absurde que le i^{e} nœud de T_R possède un sous-arbre gauche non vide. Ceci implique que la i^{e} feuille de T_R n’est pas attachée à son i^{e} nœud. Ainsi, par le lemme 4.3.2, la i^{e} feuille de T_R est attachée à son $(i-1)^{\text{e}}$ nœud et est orientée vers la droite. Dans T_L , la i^{e} feuille ne peut pas être attachée à son i^{e} nœud car $A \neq \perp$. Par conséquent, par le lemme 4.3.2, la i^{e} feuille de T_L est également attachée à son $(i-1)^{\text{e}}$ nœud et est orientée vers la droite. Comme T possède au moins i nœuds, il possède au moins $i+1$ feuilles, ce qui implique que la i^{e} feuille n’est pas la feuille la plus à droite de T_L et T_R , et ainsi, (T_L, T_R) n’est pas un couple d’arbres binaires jumeaux, ce qui contredit l’hypothèse.

Supposons maintenant que le i^{e} nœud de T_R possède un sous-arbre droit non vide. Ceci implique que la $(i+1)^{\text{e}}$ feuille de T_R n’est pas attachée à son i^{e} nœud, et donc, par le lemme 4.3.2, la $(i+1)^{\text{e}}$ feuille de T_R est orientée vers la gauche. De plus, comme le i^{e} nœud de T_R possède un sous-arbre droit non vide et que le i^{e} nœud de T_L est sa racine, le i^{e} nœud de T_L possède également un sous-arbre droit non vide. Ceci implique que la $(i+1)^{\text{e}}$ feuille de T_L n’est pas attachée à son i^{e} nœud, et donc, par le lemme 4.3.2, la $(i+1)^{\text{e}}$ feuille de T_R est aussi orientée vers la gauche. Ceci contredit le fait que (T_L, T_R) est un couple d’arbres binaires jumeaux, et implique que le i^{e} nœud de T_R n’a pas de fils.

Le cas $B \neq \perp$ se traite de la même manière. \square

Proposition 4.3.11. *Pour tout couple d’arbres binaires de recherche jumeaux (T_L, T_R) , l’algorithme EXTRAIREBAXTER calcule un mot qui appartient à la classe de Baxter encodée par (T_L, T_R) . De plus, si (T_L, T_R) est étiqueté par une permutation, le mot retourné est une permutation de Baxter.*

Démonstration. Montrons par récurrence sur la taille n de (T_L, T_R) que si (T_L, T_R) est un couple d'arbres binaires de recherche jumeaux, alors EXTRAIREBAXTER retourne un mot qui est à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$, *i.e.*, comme expliqué précédemment, un mot appartenant à la classe de Baxter encodée par (T_L, T_R) . Cette propriété se vérifie pour $n \leq 1$. Supposons maintenant que $T_L = A \wedge_a B$. Comme T_L et T_R sont des arbres binaires de recherche et sont étiquetés par un même mot, leur i^e nœud possède la même étiquette. Ainsi, par le lemme 4.3.10, il existe dans T_R un nœud x sans fils et étiqueté par a . De plus, la canopée de T_L est de la forme $v01w$ où $v := \text{cnp}(A)$, $w := \text{cnp}(B)$, et la canopée de T_R est de la forme $v'10w'$ où v' (resp. w') est le complémentaire de v (resp. w) étant donné que (T_L, T_R) est un couple d'arbres binaires jumeaux. Nous avons maintenant deux cas.

Si x est un fils gauche dans T_R , l'algorithme retourne le mot au où u est le mot obtenu en appliquant l'algorithme sur (T'_L, T'_R) où $T'_L = A / B$, et T'_R est obtenu à partir de T_R en lui supprimant le nœud x . En premier lieu, la canopée de T'_L est de la forme $v0w$, et celle de T'_R de la forme $v'1w'$. Comme T'_L et T'_R sont encore des arbres binaires de recherche, (T'_L, T'_R) est bien un couple d'arbres binaires de recherche jumeaux. Et ainsi, par hypothèse de récurrence, le mot u appartient à la classe de Baxter encodée par (T'_L, T'_R) , ce qui montre que au appartient à la classe encodée par (T_L, T_R) puisque au est une extension linéaire à la fois de $\Delta(T_L)$ et de $\nabla(T_R)$. Le cas où x est un fils droit dans T_R se traite de la même manière.

Finalement, dans le cas où (T_L, T_R) est étiqueté par une permutation, l'algorithme EXTRAIREBAXTER coïncide avec un algorithme de Dulucq et Guibert [DG94] pour lequel ces derniers ont montré qu'il retourne une permutation de Baxter. \square

La validité de l'algorithme EXTRAIREBAXTER a pour conséquence les deux résultats suivants :

Théorème 4.3.12. *Pour tout $n \geq 0$, il existe une bijection entre l'ensemble des classes d'équivalence de Baxter et l'ensemble des couples d'arbres binaires de recherche jumeaux de taille n .*

Démonstration. Par la proposition 4.3.4 et le théorème 4.3.5, l'algorithme du \mathbb{P} -symbole définit une injection entre l'ensemble des classes de Baxter et l'ensemble des couples d'arbres binaires de recherche jumeaux. De plus, par la proposition 4.3.11, l'algorithme EXTRAIREBAXTER exhibe une surjection entre ces deux ensembles. Ils sont par conséquent en bijection. \square

Le théorème 4.3.12 implique en particulier que les classes de Baxter de permutations de taille n sont en bijection avec l'ensemble des couples d'arbres binaires de recherche jumeaux étiquetés par une permutation — ou de manière équivalente, avec l'ensemble des couples d'arbres binaires jumeaux non étiquetés.

Théorème 4.3.13. *Pour tout $n \geq 0$, tout classe d'équivalence de $\mathfrak{S}_n / \equiv_B$ contient exactement une permutation de Baxter.*

Démonstration. Soit C une classe de $\mathfrak{S}_n / \equiv_B$. Par le théorème 4.3.12, C peut être encodée par un couple d'arbres binaires jumeaux J . Par la proposition 4.3.11, l'algorithme EXTRAIREBAXTER calcule une permutation appartenant à la classe de Baxter encodée par J , impliquant que chaque classe de Baxter contient au minimum une permutation de Baxter. Le théorème suit du fait qu'il y a autant de permutations de Baxter que de couples d'arbres binaires jumeaux. \square

Extraction des mots minimaux et maximaux

Reading définit dans [Rea05] les *permutations de Baxter tordues*, qui sont les permutations qui évitent les motifs généralisés $2-41-3$ et $3-41-2$. Ces permutations sont les permutations minimales pour l'ordre du permutoèdre de nos classes de Baxter de permutations. En effet,

soit σ la permutation minimale dans sa classe de Baxter. Il est alors impossible de réaliser une réécriture de la forme

$$\mathbf{x} u \mathbf{d} \mathbf{a} v \mathbf{y} \rightarrow \mathbf{x} u \mathbf{a} \mathbf{d} v \mathbf{y}, \quad (4.3.8)$$

où $\mathbf{a} < \mathbf{x}, \mathbf{y} < \mathbf{d}$ et $u, v \in A^*$. Ainsi, σ évite les motifs $2 - 41 - 3$ et $3 - 41 - 2$. Réciproquement, si σ est une permutation de Baxter tordue, elle évite les motifs $2 - 41 - 3$ et $3 - 41 - 2$ si bien qu'il est impossible d'appliquer la règle de réécriture \rightarrow dans σ , et par la proposition 4.2.9 et le lemme 4.2.10, σ est minimale de sa classe. Ceci implique que les permutations de Baxter tordues et les permutations de Baxter sont équipotentes puisque par le théorème 4.3.13, il y a exactement une permutation de Baxter par classe de Baxter de permutations, et par la proposition 4.2.9, il y a également exactement une permutation de Baxter tordue. Ceci suggère l'existence d'une bijection envoyant la permutation de Baxter d'une classe vers la permutation de Baxter tordue de cette même classe.

Comme le soulignèrent Law et Reading [LR10], West montra le premier, dans un travail non publié, une bijection entre les permutations de Baxter et les permutations de Baxter tordues, en utilisant la machinerie fournie par les arbres de génération (voir à ce sujet par exemple [Wes90] ou [BM03]). Dans notre contexte, ainsi que dans celui de Law et Reading [LR10], cette bijection est celle qui préserve les classes de Baxter. Nous proposons ci-dessous un algorithme pour calculer cette bijection.

Considérons l'algorithme suivant qui permet, étant donné un couple d'arbres binaires de recherche jumeaux (T_L, T_R) , de calculer le plus petit mot pour l'ordre lexicographique de la classe de Baxter qu'il encode.

Algorithme : EXTRAIREMIN.

Entrée : un couple d'arbres binaires de recherche jumeaux (T_L, T_R) .

Sortie : le plus petit mot pour l'ordre lexicographique de la classe de Baxter encodée par (T_L, T_R) .

1. Soit $u := \epsilon$ le mot vide.
2. Soit $F := T_L$ une forêt ordonnée d'arbres binaires. initialement composée de l'unique arbre T_L .
3. Tant que F n'est pas vide et que $T_R \neq \perp$:
 - (a) Soit i le plus petit entier tel que le i^{e} nœud de F est une racine et le i^{e} nœud de T_R n'a pas de fils.
 - (b) Soit \mathbf{a} l'étiquette du i^{e} nœud de T_L .
 - (c) Poser $u := u \cdot \mathbf{a}$.
 - (d) Supprimer le i^{e} nœud de F et le i^{e} nœud de T_R .
4. Retourner u .

Fin.

Notons que si l'on choisit dans l'instruction (3a) le plus grand entier i possible au lieu du plus petit, le précédent algorithme calcule le plus grand mot pour l'ordre lexicographique de la classe de Baxter encodée par (T_L, T_R) . Nous appelons cette variante EXTRAIREMAX.

La figure 4.7 montre une exécution de cet algorithme.

Proposition 4.3.14. *Pour tout couple d'arbres binaires de recherche jumeaux (T_L, T_R) , l'algorithme EXTRAIREMIN (resp. EXTRAIREMAX) calcule le plus petit (resp. le plus grand) mot pour l'ordre lexicographique de la classe d'équivalence de Baxter encodée par (T_L, T_R) . De plus, si (T_L, T_R) est étiqueté par une permutation, le mot retourné est la plus petite (resp. la plus grande) permutation de sa classe de Baxter pour l'ordre du permutoèdre.*

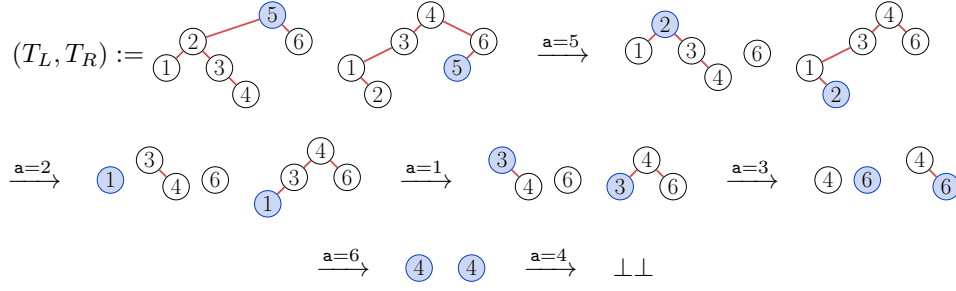


FIGURE 4.7 – Une exécution de l’algorithme EXTRAIREMIN sur le couple d’arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation calculée est 521364 et est le plus petit élément de sa classe de Baxter.

Démonstration. La sortie u de l’algorithme EXTRAIREMIN est clairement à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$. Ceci implique que u appartient à la classe de Baxter encodée par (T_L, T_R) . De plus, l’exécution de cet algorithme se termine puisque par le théorème 4.3.13, pour tout couple d’arbres binaires de recherche jumeaux, il existe au moins un mot qui est à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$. La minimalité (resp. maximalité) de la permutation calculée provient de fait qu’à chaque étape, le nœud qui possède la plus petite (resp. la plus grande) étiquette est choisi.

Finalement, étant donné que l’ordre lexicographique sur les permutations est une extension linéaire de l’ordre du permutoèdre, et que d’après la proposition 4.2.9, les classes d’équivalence de Baxter de permutations sont des intervalles du permutoèdre, l’algorithme EXTRAIREMIN (resp. EXTRAIREMAX) retourne la plus petite (resp. la plus grande) permutation de sa classe pour l’ordre du permutoèdre. \square

D’après la proposition 4.3.14, et en utilisant notre analogue de l’algorithme d’insertion de Robinson-Schensted, nous pouvons calculer la bijection entre les permutations de Baxter et les permutations de Baxter tordues de la façon suivante. Si σ est une permutation de Baxter, en appliquant EXTRAIREMIN sur $\mathbb{P}(\sigma)$, nous obtenons la permutation de Baxter tordue en bijection avec σ . Réciproquement, si σ est une permutation de Baxter tordue, en appliquant EXTRAIREBAXTER sur $\mathbb{P}(\sigma)$, nous obtenons la permutation de Baxter en bijection avec σ .

De manière similaire, nous pouvons calculer simplement une bijection entre les permutations de Baxter et les permutations qui évitent les motifs $2-14-3$ et $3-14-2$. Ces permutations sont les plus grands éléments de leur classe de Baxter, et la bijection est analogue à celle que nous venons de décrire en utilisant cette fois-ci l’algorithme EXTRAIREMAX.

4.3.4 Version itérative de l’algorithme d’insertion

Dans cette partie, nous proposons une version itérative du \mathbb{P} -symbole de Baxter, ou en d’autres termes, un algorithme d’insertion \leftarrow tel que pour tout mot u vérifiant $\mathbb{P}(u) = (T_L, T_R)$, l’insertion d’une lettre a dans le couple d’arbres binaires de recherche jumeaux (T_L, T_R) vérifie $(T_L, T_R) \leftarrow a = \mathbb{P}(ua)$.

Ceci — additionnellement au fait d’être en accord avec l’algorithme classique d’insertion de Robinson-Schensted qui lui-même est itératif — donne un moyen de calculer dans le monoïde de Baxter. En effet, rappelons que d’après le théorème 4.3.5, toute classe de Baxter peut s’encoder par un couple d’arbres binaires de recherche jumeaux. Ainsi, pour concaténer deux classes de Baxter dans le monoïde de Baxter, ou de manière équivalente, deux couples d’arbres binaires de recherche jumeaux (T_L, T_R) et (T'_L, T'_R) , il suffit de calculer un mot u' appartenant à la classe

encodée par (T'_L, T'_R) , de manière indifférente avec l'algorithme EXTRAIREBAXTER, EXTRAIREMIN, ou EXTRAIREMAX puis d'insérer les lettres de u' de la gauche vers la droite dans (T_L, T_R) .

Nous avons besoin de quelques définitions avant de poser cet algorithme. Soit T un arbre binaire de recherche droit et \mathbf{b} une lettre. La *restriction de T aux valeurs inférieures à \mathbf{b}* , noté $T_{\leq \mathbf{b}}$, est l'arbre binaire de recherche droit composé uniquement des nœuds x de T étiquetés par des lettres \mathbf{a} vérifiant $\mathbf{a} \leq \mathbf{b}$, et tel que pour tous nœuds x et y de $T_{\leq \mathbf{b}}$, si x est un ancêtre de y dans $T_{\leq \mathbf{b}}$, alors x est aussi un ancêtre de y dans T . De manière analogue, on définit la *restriction de T aux valeurs strictement supérieures à \mathbf{b}* , noté $T_{> \mathbf{b}}$. La figure 4.8 illustre ces définitions.

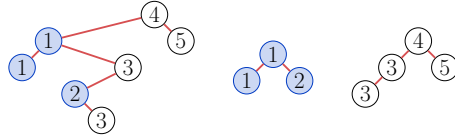


FIGURE 4.8 – Un arbre binaire de recherche droit T et les arbres binaires $T_{\leq 2}$ et $T_{> 2}$.

L'*insertion en racine* d'une lettre \mathbf{a} dans T consiste à modifier T de sorte que sa racine soit un nouveau nœud étiqueté par \mathbf{a} , son sous-arbre gauche $T_{\leq \mathbf{a}}$, et son sous-arbre droit $T_{> \mathbf{a}}$.

Nous pouvons maintenant énoncer les versions itératives des \mathbb{P} et \mathbb{Q} -symboles de Baxter :

Définition 4.3.15. Soit (T_L, T_R) un couple d'arbres binaires de recherche jumeaux et \mathbf{a} une lettre. L'insertion de \mathbf{a} dans (T_L, T_R) consiste à insérer en feuille \mathbf{a} dans T_L , et à insérer en racine \mathbf{a} dans T_R . Le \mathbb{P} -symbole itératif de Baxter d'un mot u est le couple d'arbres binaires obtenu en insérant successivement les lettres de u de gauche à droite dans le couple (\perp, \perp) . Le \mathbb{Q} -symbole itératif de Baxter de u est le couple d'arbres binaires jumeaux (S_L, S_R) où les nœuds de S_L (resp. S_R) sont étiquetés par leur moment de création dans T_L (resp. T_R).

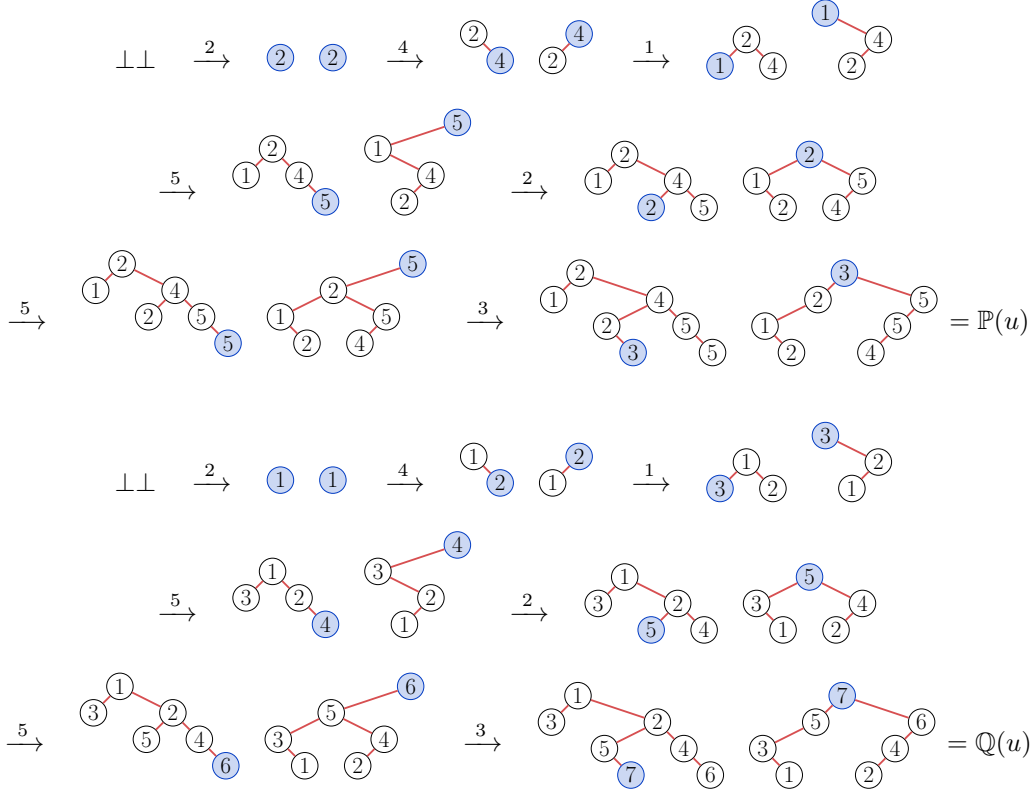
La figure 4.9 illustre, étape par étape, un calcul des \mathbb{P} et \mathbb{Q} -symboles itératifs de Baxter d'un mot.

Pour montrer que le \mathbb{P} -symbole itératif fourni par la définition 4.3.15 construit le même couple d'arbres binaires de recherche jumeaux que le \mathbb{P} -symbole de la définition 4.3.1, montrons le lemme qui suit.

Lemme 4.3.16. Soit u un mot, T l'arbre binaire de recherche droit obtenu en insérant en racine les lettres de u de la gauche vers la droite, et T' l'arbre binaire de recherche droit obtenu en insérant en feuille les lettres de u de la droite vers la gauche. Alors, $T = T'$.

Démonstration. Procédons par récurrence sur $|u|$. Si $u = \epsilon$, le lemme est vérifié puisque $T = T' = \perp$. Sinon, supposons que $u = v\mathbf{a}$ avec $\mathbf{a} \in A$. Soit S l'arbre binaire de recherche droit obtenu par insertion en racine des lettres de v , lu de la gauche vers la droite. Par hypothèse de récurrence, S est aussi l'arbre binaire de recherche droit obtenu par des insertions en feuille des lettres de v lu de la droite vers la gauche. Par définition, T vérifie $T = S_{\leq \mathbf{a}} \wedge_{\mathbf{a}} S_{> \mathbf{a}}$, et T' est de la forme $T' = L' \wedge_{\mathbf{a}} R'$ où le sous-arbre L' dépend uniquement du sous-mot $v_{\leq \mathbf{a}} := v_{] -\infty, \mathbf{a}]}$ et le sous-arbre R' dépend uniquement du sous-mot $v_{> \mathbf{a}} := v_{] \mathbf{a}, +\infty[}$. Ainsi, par hypothèse de récurrence, $L' = S_{\leq \mathbf{a}}$ et $R' = S_{> \mathbf{a}}$, montrant $T = T'$. \square

Proposition 4.3.17. Pour tout mot u , le \mathbb{P} -symbole de Baxter de u et le \mathbb{P} -symbole itératif de u sont égaux.

FIGURE 4.9 – Les étapes de calcul du \mathbb{P} et du \mathbb{Q} -symbole de Baxter de $u := 2415253$.

Démonstration. Soit (T_L, T_R) le \mathbb{P} -symbole de u et (T'_L, T'_R) le \mathbb{P} -symbole itératif de u . Par définition de ces deux algorithmes d'insertion, on a $T_L = T'_L$. De plus, T_R est obtenu par des insertions en feuille des lettres de u lues de la droite vers la gauche, et T'_R est obtenu par des insertions en racine des lettres de u de la gauche vers la droite. Par le lemme 4.3.16, $T_R = T'_R$. \square

La proposition 4.3.17 implique aussi que le \mathbb{Q} -symbole de Baxter et le \mathbb{Q} -symbole itératif de Baxter de tout mot u sont les mêmes.

4.4 Un treillis sur les couples d'arbres binaires jumeaux

Lorsqu'une relation d'équivalence sur les éléments d'un treillis possède certaines propriétés, celle-ci permet de définir un treillis quotient. Nous rappelons la définition des congruences de treillis et ensuite, montrons que la relation de Baxter, qui est une congruence de monoïde, est aussi une congruence de treillis du permutoèdre. De cela, nous en déduisons une structure de treillis sur l'ensemble des couples d'arbres binaires jumeaux. Nous caractérisons ses relations de couverture puis introduisons un nouvel objet combinatoire en bijection avec les permutations de Baxter, à savoir les diagrammes de Tamari doubles, qui se révèlent être adaptés pour tester si deux éléments sont comparables dans le treillis des couples arbres binaires jumeaux.

Remarquons que notre treillis des couples d'arbres binaires jumeaux est isomorphe au treillis obtenu par Reading en [Rea05] lorsqu'il considère le quotient du permutoèdre par sa congruence de treillis. Une étude complémentaire de ce treillis — dont les objets sont vus comme des quadrangulations — fut réalisée en [LR10].

4.4.1 Congruences de treillis

Si I est un intervalle d'un poset, nous notons $I\uparrow$ (resp. $I\downarrow$) le plus petit (resp. le plus grand) élément de I .

Rappelons la définition suivante introduite en [CS98], que nous adaptons ici dans le contexte des treillis finis.

Définition 4.4.1. Soit (L, \leq) un treillis fini et \equiv une relation d'équivalence sur L . L'équivalence \equiv est également une congruence de treillis si les trois conditions suivantes sont vérifiées :

- (i) toute classe d'équivalence de \equiv est un intervalle de L ;
- (ii) pour tous $x, y \in L$, si $x \leq y$, alors $\hat{x}\uparrow \leq \hat{y}\uparrow$, où \hat{x} et \hat{y} sont les classes d'équivalence respectives de x et de y ;
- (iii) pour tous $x, y \in L$, si $x \leq y$, alors $\hat{x}\downarrow \leq \hat{y}\downarrow$, où \hat{x} et \hat{y} sont les classes d'équivalence respectives de x et de y .

Chajda et Snášel [CS98] ont montré qu'une congruence de treillis permet de construire un poset de la manière suivante :

Définition 4.4.2. Soit (L, \leq) un treillis fini et \equiv une congruence de treillis de L . Alors, le poset quotient de L par \equiv est l'ensemble L/\equiv muni de la relation d'ordre \leq_{\equiv} définie pour toutes classes d'équivalence \hat{x} et \hat{y} de \equiv par

$$\hat{x} \leq_{\equiv} \hat{y} \quad \text{si et seulement si} \quad \text{il existe } x \in \hat{x} \text{ et } y \in \hat{y} \text{ tels que } x \leq y. \quad (4.4.1)$$

Il s'avère de plus que le quotient d'un treillis est aussi un treillis (voir par exemple [Rea04] et [Rea05]), ce qui rend cette définition particulièrement agréable puisqu'elle permet d'obtenir de nouveaux treillis à partir d'anciens. Comme nous l'avons déjà mentionné, il est naturel d'observer le comportement d'une congruence de monoïde d'un bon monoïde en la restreignant aux permutations, et, par le fait que le permutoèdre munit l'ensemble des permutations d'une structure de treillis, il est également naturel de se demander si la congruence de monoïde dont il est question est également une congruence de treillis. Si la réponse est positive, on obtient ainsi gratuitement une structure de treillis sur les objets combinatoires qui indexent les classes de permutations. C'est précisément ce que nous montrons dans le paragraphe suivant au sujet de la congruence de Baxter.

4.4.2 Le treillis des couples d'arbres binaires jumeaux

Si σ est une permutation, on note $\sigma\uparrow$ (resp. $\sigma\downarrow$) le plus petit (resp. le plus grand) élément de sa classe de Baxter. Rappelons que l'existence de ces deux éléments est assurée par la proposition 4.2.9.

Une congruence du permutoèdre

Théorème 4.4.3. La relation d'équivalence de Baxter est une congruence de treillis du permutoèdre.

Démonstration. D'après la proposition 4.2.9, toute classe de Baxter de permutations est bien un intervalle du permutoèdre. Il reste donc à montrer les deux derniers points de la définition 4.4.1.

Soient σ et ν deux permutations telles que $\sigma \leq_p \nu$. Montrons alors que $\sigma\downarrow \leq_p \nu\downarrow$. Il suffit de montrer cette inégalité uniquement lorsque $\nu = \sigma \cdot s_i$ où s_i est une transposition élémentaire et i n'est pas une descente de σ . Si $\sigma = \sigma\downarrow$, alors $\sigma\downarrow \leq_p \nu \leq_p \nu\downarrow$ et la propriété est établie. Autrement, par le lemme 4.2.10, il existe une transposition élémentaire s_j et une permutation

π tels que π et σ sont adjacents pour \xleftrightarrow{B} , $\pi = \sigma \cdot s_j$, et $\sigma \leq_P \pi$. Il reste à démontrer qu'il existe une permutation μ telle que $\nu \equiv_B \mu$ et $\pi \leq_P \mu$. En effet, ceci mène à montrer, en réappliquant itérativement ce raisonnement, que $\sigma \downarrow$ est plus petite pour l'ordre du permutoèdre qu'une permutation appartenant à la classe de Baxter de ν , et ainsi, par transitivité, que $\sigma \downarrow \leq_P \nu \downarrow$. Nous avons quatre cas à explorer suivant les indices i et j .

Cas 1. Si $j \leq i - 2$, σ est de la forme $\sigma = uabvcdw$ où a (resp. c) est la j^e (resp. i^e) lettre de σ et $a < b$ et $c < d$ puisque i et j ne sont pas des descentes de σ . Nous avons $\nu = uabvcdw$ et $\nu \cdot s_j = ubavcdw =: \mu$. De plus, comme $\pi \xleftrightarrow{B} \sigma$, il existe des lettres $x \in \text{Alph}(u)$ et $y \in \text{Alph}(vcdw)$ telles que $a < x, y < b$. Ainsi, $\mu \xleftrightarrow{B} \nu$. Finalement, puisque $\pi = uabvcdw$, $\pi \leq_P \mu$, de sorte que μ est la permutation recherchée.

Cas 2. le cas $j \geq i + 2$ est analogue au précédent.

Cas 3. si $j = i + 1$, σ est de la forme $\sigma = uabcv$ où a est la i^e lettre de σ et $a < b < c$ puisque i et j ne sont pas des descentes de σ . Comme $\sigma \xleftrightarrow{B} \pi$, il existe des lettres $x \in \text{Alph}(u)$ et $y \in \text{Alph}(v)$ telles que $b < x, y < c$. Ainsi, puisque $\nu = ubacv$ et $a < b < x, y < c$, on a $\nu \cdot s_j = ubcav \xleftrightarrow{B} \nu$. De plus, $\nu \cdot s_j \cdot s_i = ucba v =: \mu$ et $\nu \cdot s_j \xleftrightarrow{B} \nu \cdot s_j \cdot s_i$ puisque $b < x, y < c$ et donc, $\mu \equiv_B \nu$. Finalement, comme $\pi = uabcv$, on $\pi \leq_P \mu$, et ainsi μ est la permutation recherchée.

Cas 4. le cas $j = i - 1$ est analogue au précédent.

Le cas $\sigma \uparrow \leq_P \nu \uparrow$ se traite de la même manière. □

Le treillis de Baxter

Rappelons que par le théorème 4.3.12, toute classe de Baxter de permutations de taille n peut s'encoder par un couple d'arbres binaires jumeaux. De ce fait, et par la définition 4.4.2, le quotient du permutoèdre d'ordre n par la relation de Baxter est le treillis de la forme (\mathcal{J}_n, \leq_B) où la relation d'ordre \leq_B vérifie, pour tout $J_0, J_1 \in \mathcal{J}_n$,

$$J_0 \leq_B J_1 \quad \text{si et seulement si} \quad \begin{array}{l} \text{il existe } \sigma, \nu \in \mathfrak{S}_n \text{ tels que} \\ \sigma \leq_P \nu, \mathbb{P}(\sigma) = J_0 \text{ et } \mathbb{P}(\nu) = J_1. \end{array} \quad (4.4.2)$$

Nous appelons ce treillis le *treillis de Baxter*. La figure 4.10 montre les classes de Baxter dans le permutoèdre d'ordre 4.

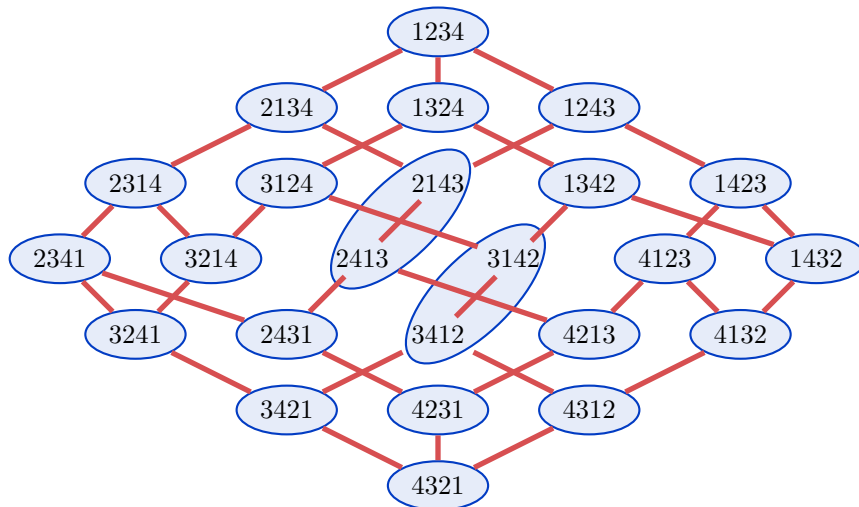


FIGURE 4.10 – Le permutoèdre d'ordre 4 découpé en classes d'équivalence de Baxter.

Relations de couvertures

Décrivons maintenant ses relations de couverture. Considérons pour cela une classe $\hat{\sigma}$ de Baxter de permutations encodée par le couple d'arbres binaires jumeaux (T_L, T_R) , et σ le plus grand élément de $\hat{\sigma}$. Pour toute descente en position i de σ , la permutation $\sigma \cdot s_i$ n'est plus dans $\hat{\sigma}$, et, par définition du treillis de Baxter, le couple d'arbres binaires jumeaux $\mathbb{P}(\sigma \cdot s_i) =: (T'_L, T'_R)$ couvre (T_L, T_R) . Les permutations σ et $\sigma \cdot s_i$ sont de la forme

$$\sigma = u \mathbf{ad} v \quad \text{et} \quad \sigma \cdot s_i = u \mathbf{da} v, \quad (4.4.3)$$

où $\mathbf{a} < \mathbf{d}$. Il y a trois cas différents suivant si le facteur u ou v possède une lettre intermédiaire en valeur entre \mathbf{a} et \mathbf{d} . En se basant sur le fait que la relation sylvestre est une congruence du permutoèdre, que celle-ci permet de définir le treillis de Tamari (voir la démonstration dans [HNT05]), et que les couvertures de ce dernier sont des rotations (voir la définition 1.3.1 du chapitre 1), nous trouvons les couvertures suivantes dans le treillis de Baxter :

1. s'il existe une lettre y dans v telle que $\mathbf{a} < y < \mathbf{d}$, alors $T'_R = T_R$ et T'_L est obtenu en réalisant une rotation gauche dans T_L qui ne modifie pas sa canopée ;
2. s'il existe une lettre x dans u telle que $\mathbf{a} < x < \mathbf{d}$, alors $T'_L = T_L$ et T'_R est obtenu en réalisant une rotation droite dans T_R qui ne modifie pas sa canopée ;
3. si aucune lettre de u ni de v n'est comprise en valeur entre \mathbf{a} et \mathbf{d} , alors T'_L (resp. T'_R) est obtenu en réalisant une rotation gauche (resp. rotation droite) dans T_L (resp. T_R) qui modifie sa canopée.

Ainsi, d'après cette caractérisation des relations de couverture du treillis de Baxter et la définition de l'ordre de Tamari (voir la définition 1.3.2 du chapitre 1), nous avons, pour tous couples d'arbres binaires jumeaux (T_L, T_R) et (T'_L, T'_R) ,

$$(T_L, T_R) \leqslant_{\mathbf{B}} (T'_L, T'_R) \quad \text{si et seulement si} \quad T'_L \leqslant_{\mathbf{T}} T_L \text{ et } T_R \leqslant_{\mathbf{T}} T'_R. \quad (4.4.4)$$

Notons qu'une rotation droite en y réalisée dans un arbre binaire T modifie sa canopée si et seulement si le sous-arbre droit B du fils gauche x de y est vide (voir la figure 1.4 du chapitre 1). Similairement, une rotation gauche en y modifie la canopée de T si et seulement si le sous-arbre gauche de B de y est vide. De plus, si y est le i^{e} nœud de T , par le lemme 4.3.2, nous pouvons observer que B est aussi la i^{e} feuille de T . Par conséquent, une rotation droite (resp. gauche) en y modifie l'orientation de la i^{e} feuille de T de la droite vers la gauche (resp. de la gauche vers la droite).

4.4.3 Diagrammes de Tamari doubles

Décider si deux couples d'arbres binaires jumeaux sont comparables dans le treillis de Baxter n'est pas chose évidente. Le but de ce paragraphe est d'introduire un nouvel objet combinatoire en bijection avec les couples d'arbres binaires jumeaux. Celui-ci permet de réaliser de manière transparente les comparaisons dans le treillis de Baxter.

Diagrammes de Tamari

Nous nous basons sur le fait qu'il existe un moyen de comparer facilement deux arbres binaires dans le treillis de Tamari en utilisant l'objet combinatoire suivant :

Définition 4.4.4. *Un diagramme de Tamari est un mot u sur l'alphabet \mathbb{N} qui respecte les deux conditions suivantes :*

- (i) $0 \leqslant u_i \leqslant n - i$ pour tout $i \in [n]$;

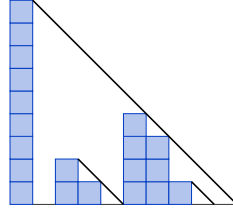


FIGURE 4.11 – Le diagramme de Tamari 9021043100 dans sa représentation graphique et ses diagonales.

(ii) $u_{i+j} \leq u_i - j$ pour tout $i \in [n]$ et $j \in [u_i]$.

La taille d'un diagramme de Tamari est son nombre de lettres.

Un diagramme de Tamari se représente avantageusement sous une forme graphique, en représentant chaque lettre par des colonnes dont les hauteurs dépendent des valeurs des lettres. Vérifier la condition (ii) devient alors très simple puisqu'il suffit de tracer du sommet de chaque colonne une diagonale de pente $-\frac{1}{2}$ comme indiqué dans la figure 4.11, et de s'assurer qu'aucune colonne n'en dépasse. De même, on vérifie la condition (i) en s'assurant qu'aucune diagonale ne tombe à droite du diagramme.

Pallo [Pal86] montra que l'ensemble des diagrammes de Tamari de taille n est en bijection avec l'ensemble des arbres binaires à n nœuds. Cette bijection se décrit de la manière suivante. On associe à un arbre binaire T le diagramme de Tamari $\text{dt}(T)$ obtenu en étiquetant chaque nœud x de T par le nombre de nœuds du sous-arbre droit de x et considérant le mot obtenu par sa lecture infixe. Réciproquement, si u est un diagramme de Tamari de taille n , on lui associe un arbre binaire $\text{dt}^{-1}(u)$ récursivement de la manière suivante :

$$\text{dt}^{-1}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{dt}^{-1}(u_1 \dots u_{i-1}) \wedge \text{dt}^{-1}(u_{i+1} \dots u_n) & \text{où } i := \min\{i : u_i = n - i\} \text{ sinon.} \end{cases} \quad (4.4.5)$$

Les diagrammes de Tamari sont de bonnes manières de voir les sommets du treillis de Tamari puisque l'ordre s'y lit naturellement [Knu04]. En effet, si T et T' sont deux arbres binaires à n nœuds, nous avons

$$T \leq_T T' \quad \text{si et seulement si} \quad \text{dt}(T)_i \leq \text{dt}(T')_i \text{ pour tout } i \in [n]. \quad (4.4.6)$$

Diagrammes de Tamari doubles

Si u est un mot de \mathbb{Z}^* , nous notons par $u \uparrow$ l'image de u par le morphisme de monoïde qui envoie toute lettre $a \in \mathbb{Z}$ vers $\max\{0, a\}$. De même, on note $u \downarrow$ l'image de u par le morphisme de monoïde qui envoie toute lettre $a \in \mathbb{Z}$ vers $-\min\{0, a\}$. Par exemple, pour $u := (2, -2, -1, 3, 1, -1, 0)$, nous avons $u \uparrow = (2, 0, 0, 3, 1, 0, 0)$ et $u \downarrow = (0, 2, 1, 0, 0, 1, 0)$.

Définition 4.4.5. Un diagramme de Tamari double $\delta = \delta_1 \dots \delta_n$ est un mot de \mathbb{Z}^* où

- (i) $\delta \uparrow$ et $\delta \downarrow$ sont des diagrammes de Tamari;
- (ii) pour tout $i \in [n-1]$, $\delta_i \neq 0$.

La taille d'un diagramme de Tamari double est son nombre de lettres.

Nous représentons graphiquement un diagramme de Tamari double δ en dessinant, pour chaque lettre δ_i une colonne de hauteur correspondante orientée vers le haut si δ_i est positive

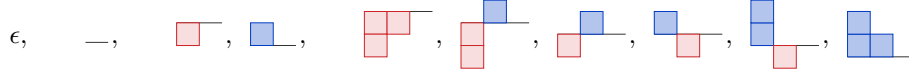


FIGURE 4.12 – Les diagrammes de Tamari doubles de tailles 0, 1, 2 et 3.

ou vers vers le bas sinon. Les premiers diagrammes de Tamari doubles sont représentés suivant cette convention dans la figure 4.12.

Montrons que les diagrammes de Tamari doubles sont des objets de la famille combinatoire de Baxter :

Proposition 4.4.6. *Pour tout $n \geq 0$, l'ensemble des diagrammes de Tamari doubles de taille n est en bijection avec l'ensemble des couples d'arbres binaires jumeaux à n nœuds. De plus, cette bijection se calcule de la manière suivante. Si $J := (T_L, T_R)$ est un couple d'arbres binaires jumeaux, le diagramme de Tamari double $\text{dtd}(J)$ qui lui est associé est défini par*

$$\text{dtd}(J)_i := \begin{cases} u_i & \text{si } u_i \neq 0 \text{ et } v_i = 0, \\ -v_i & \text{si } v_i \neq 0 \text{ et } u_i = 0, \\ 0 & \text{sinon } (i = n), \end{cases} \quad (4.4.7)$$

où $u := \text{dt}(T_R)$ et $v := \text{dt}(T_L)$. Inversement, si δ est un diagramme de Tamari double, on lui associe le couple d'arbres binaires jumeaux

$$\text{dtd}^{-1}(\delta) := (\text{dt}^{-1}(\delta \downarrow), \text{dt}^{-1}(\delta \uparrow)). \quad (4.4.8)$$

Démonstration. Commençons par montrer que (4.4.7) définit bien un diagramme de Tamari double. Soient pour cela $J := (T_L, T_R)$ un couple d'arbres binaires jumeaux de taille n , $u := \text{dt}(T_R)$, $v := \text{dt}(T_L)$ et $\delta := \text{dtd}(J)$. Pour que δ soit bien défini, il faut d'une part que $u_i = v_i = 0$ implique $i = n$, et d'autre part que l'on ait $u_i = 0$ ou $v_i = 0$ pour tout $i \in [n]$. Supposons qu'il existe $i \in [n]$ tel que $u_i = v_i = 0$. Par définition des diagrammes de Tamari, ceci implique que les i^{e} nœuds de T_L et de T_R n'ont pas de fils droit. Ainsi, par le lemme 4.3.2, les $(i+1)^{\text{e}}$ feuilles de T_L et de T_R sont attachées aux i^{e} nœuds de T_L et de T_R et sont orientées vers la droite. Comme J est un couple d'arbres binaires jumeaux, la $(i+1)^{\text{e}}$ feuille est nécessairement la feuille la plus à droite des deux arbres, et ainsi, $i = n$. Pour montrer la seconde condition, supposons par l'absurde qu'il existe $i \in [n]$ tel que $u_i \neq 0$ et $v_i \neq 0$. Par définition des diagrammes de Tamari, ceci implique que les i^{e} nœuds de T_L et de T_R ont tous deux un fils droit non vide. Ainsi, par le lemme 4.3.2, les $(i+1)^{\text{e}}$ feuilles sont nécessairement attachées aux $(i+1)^{\text{e}}$ nœuds de T_L et de T_R , et orientées vers la gauche. Ceci est en contradiction avec le fait que J est un couple d'arbres binaires jumeaux. Le mot δ est ainsi bien défini et est un diagramme de Tamari double.

Inversement, montrons que si δ est un diagramme de Tamari double, le couple $(T_L, T_R) := \text{dtd}^{-1}(\delta)$ est bien un couple d'arbres binaires jumeaux. Par l'absurde, supposons que les i^{e} feuilles de T_L et de T_R sont orientées vers la gauche avec $i \geq 2$. D'après le lemme 4.3.2, ceci implique que les sous-arbres droits des $(i-1)^{\text{e}}$ nœuds de T_L et de T_R sont non vides. Ceci contredit le fait que δ est un diagramme de Tamari double. De même, si l'on suppose par l'absurde que les i^{e} feuilles de T_L et de T_R sont orientées vers la droite avec $i \leq n-1$, alors, par le lemme 4.3.2, ces feuilles sont toutes deux attachées aux $(i-1)^{\text{e}}$ nœuds de T_L et T_R . Ceci contredit également le fait que δ est un diagramme de Tamari double.

On conclut finalement par le fait que l'application dt qui à un arbre binaire associe un diagramme de Tamari est une bijection [Pal86]. \square

Nous avons maintenant le moyen suivant de comparer deux éléments dans le treillis de Baxter :

Proposition 4.4.7. *Soient $J := (T_L, T_R)$ et $J' := (T'_L, T'_R)$ deux couples d'arbres binaires jumeaux de taille n . Nous avons alors, dans le treillis de Baxter,*

$$(T_L, T_R) \leq_B (T'_L, T'_R) \quad \text{si et seulement si} \quad \text{dtd}(J)_i \leq \text{dtd}(J')_i \quad \text{pour tout } i \in [n]. \quad (4.4.9)$$

Démonstration. Le résultat suit de caractérisation de la relation d'ordre du treillis de Baxter fournie en (4.4.4), du procédé de comparaison de deux arbres binaires dans le treillis de Tamari donné par (4.4.6) en passant par leurs diagrammes de Tamari et de l'expression de la bijection dtd fournie par la proposition 4.4.6 entre les couples d'arbres binaires jumeaux et les diagrammes de Tamari doubles. \square

La figure 4.13 montre un intervalle du treillis de Baxter, où les sommets sont vus comme des couples d'arbres binaires jumeaux, ou de manière équivalente, comme des diagrammes de Tamari doubles.

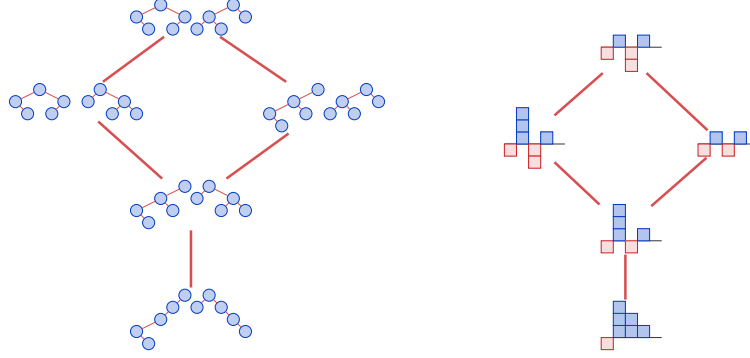


FIGURE 4.13 – Un intervalle du treillis de Baxter d'ordre 5 dont les sommets sont interprétés comme des couples d'arbres binaires jumeaux et comme des diagrammes de Tamari doubles.

Chapitre 5

L'algèbre de Hopf des couples d'arbres binaires jumeaux

Dans ces dernières années, un grand nombre d'algèbres de Hopf combinatoires dont les bases sont indexées par divers objets combinatoires plus ou moins exotiques ont été activement étudiées. Par exemple, l'algèbre de Hopf des permutations, initialement définie par Malvenuto et Reutenauer [MR95] et appelée algèbre de Hopf de fonctions quasi-symétriques libres ou encore **FQSym**, a donné lieu à de nombreuses publications [Mal93], [DHT02], [LR02], [AS05], [DHNT08] (voir aussi le paragraphe 2.2.3 du chapitre 2 où l'on rappelle la définition de cette algèbre de Hopf et quelques-unes de ses propriétés que nous utiliserons dans la suite). Celle-ci est intéressante à bien des points de vue et notamment parce qu'elle admet un grand nombre de sous-algèbres de Hopf. Citons par exemple l'algèbre de Hopf **FSym** des fonctions symétriques libres [PR95], [DHT02] basée sur les tableaux de Young standard, l'algèbre de Hopf **Bell** [Rey07] basée sur les partitions d'ensemble, l'algèbre de Hopf **PBT** de Loday-Ronco [LR98], [HNT05] basée sur les arbres binaires, ainsi que l'algèbre de Hopf **Sym** des fonctions symétriques non commutatives [GKL⁺95] basée sur les compositions d'entiers. Une approche unificatrice pour construire toutes ces structures débute par la définition d'un bon monoïde (voir la définition 4.1.4 du chapitre 4). En effet, **FSym** s'obtient directement à partir du monoïde plaxique [LS81], **Bell** à partir du monoïde de Bell [Rey07], **PBT** à partir du monoïde sylvestre [HNT02], [HNT05] et **Sym** à partir du monoïde hypoplaxique [KT97], [Nov98]. La richesse de cette approche tient au fait que, en plus d'obtenir des algèbres de Hopf, cette construction apporte au passage la plupart du temps des posets, des algorithmes combinatoires et des analogues d'algorithmes d'insertion et de correspondance de Robinson-Schensted, d'intérêts propres.

L'objectif premier de ce chapitre est d'enrichir cette collection de sous-algèbres de Hopf de **FQSym** par l'introduction d'une algèbre de Hopf basée sur les couples d'arbres binaires jumeaux (voir la définition 4.2.1 du chapitre 4). Pour ce faire, nous décrivons en détail le procédé de construction formalisé par Hivert et Nzeutchap [HN07] qui, à un bon monoïde, associe une sous-algèbre de Hopf de **FQSym**. Nous présentons, en tant que résultats intermédiaires, des propriétés générales des algèbres de Hopf ainsi construites. L'algèbre de Hopf que nous obtenons découle de la définition du monoïde de Baxter (dont l'introduction et l'étude figurent dans le chapitre 4) et est étudiée sous la loupe des propriétés que nous avons dégagées.

Dans le but d'illustrer par des exemples les relations existantes entre les congruences de treillis (voir la définition 4.4.1 du chapitre 4) et les algèbres de Hopf, Reading présenta dans [Rea05] une congruence du permutoèdre dont les classes d'équivalence sont indexées par les permutations de Baxter tordues (la définition de ces permutations figure dans le paragraphe 4.3.3 du chapitre 4). Cette congruence est très naturelle : la borne inférieure de deux congruences du permutoèdre

impliquées dans la construction de **PBT** peut être considérée comme l’un des points de départ dans la construction de **Sym** ; une question naturelle est donc de comprendre ce qu’il se passe lorsque c’est la borne supérieure qui est considérée. Reading montra que sa congruence du permutoèdre est précisément cette dernière et que les plus petits éléments de ses classes d’équivalence sont des permutations de Baxter tordues. En outre, et grâce à sa théorie, il obtint gratuitement une algèbre de Hopf basée sur les permutations de Baxter tordues. En réalité, les permutations de Baxter et les permutations de Baxter tordues sont en bijection (voir le paragraphe 4.3.3 du chapitre 4). Law et Reading [LR10] mentionnèrent que la première démonstration de cette propriété fut apportée par West dans une note non publiée. Par conséquent, l’algèbre de Hopf de Reading définie en [Rea05] peut déjà être vue comme une algèbre de Hopf sur les couples d’arbres binaires jumeaux, et notre construction en est une différente d’une même algèbre de Hopf. De plus, Law et Reading [LR10] ont très récemment détaillé leur construction et étudié quelques-unes des propriétés de cette algèbre de Hopf. Si le point de vue de Law et de Reading mène comme le nôtre globalement aux mêmes résultats, notre cheminement est différent, ce qui offre différentes manières de comprendre cette construction, l’une centrée sur la théorie des treillis, l’autre axée sur la combinatoire des mots. De plus, une grande partie des résultats de ce chapitre ne se trouve pas dans [LR10], et réciproquement. Par exemple, dans notre cas, nous donnons une réalisation polynomiale de l’algèbre de Hopf des couples d’arbres binaires jumeaux, montrons qu’elle admet une structure de bigèbre bidendriforme, qu’elle est libre, autoduale et donnons deux bases multiplicatives.

Ce chapitre est organisé comme suit. Dans le paragraphe 5.1.1, nous rappelons la construction qui à un bon monoïde associe une sous-algèbre de Hopf de **FQSym**. Nous présentons aussi en guise d’exemples les constructions de **FSym**, **PBT** et **Sym**. Le paragraphe 5.2 recense des propriétés générales des algèbres de Hopf obtenues selon ce procédé. Nous montrons en particulier que lorsque la congruence utilisée pour définir un bon monoïde est aussi une congruence de treillis, le produit dans la base fondamentale de l’algèbre de Hopf associée s’exprime comme un intervalle du treillis quotient du permutoèdre. Nous proposons également dans ce cas des bases multiplicatives et montrons que la structure est libre en tant qu’algèbre. Nous introduisons ensuite le concept de base booléenne et donnons une condition nécessaire sur le bon monoïde pour que la base fondamentale de l’algèbre de Hopf associée en soit une. Nous rappelons enfin la notion de bigèbre bidendriforme due à Foissy [Foi07], [Foi09] et donnons une condition nécessaire pour que l’algèbre de Hopf construite admette une telle structure, impliquant entre autres son autodualité. Le paragraphe 5.3 est finalement consacrée à notre construction de l’algèbre des couples d’arbres binaires jumeaux ainsi qu’à son étude.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir11].

5.1 Construction de sous-algèbres de Hopf de **FQSym**

5.1.1 La construction

Nous rappelons ici la construction formalisée par Hivert et Nzeutchap [HN07] (voir également [DHT02], [Hiv04] et [Nze08]) qui établit un lien entre bons monoïdes et algèbres de Hopf combinatoires, en donnant un moyen de construire des algèbres de Hopf à partir de bons monoïdes.

Soit $M := A^*/\equiv$ un bon monoïde au sens de la définition 4.1.4 du chapitre 4. Nous noterons \bar{u} la classe d’équivalence du mot u pour la relation \equiv .

L’algèbre de Hopf $MFQSym$

On peut associer à M une algèbre de Hopf combinatoire dont les bases sont indexées par les classes d’équivalences de \equiv composées de permutations. Cette algèbre de Hopf est une sous-

algèbre de Hopf de **FQSym**, et ses éléments de base $\mathbf{P}_{\hat{\sigma}}$ sont définis par

$$\mathbf{P}_{\hat{\sigma}} := \sum_{\sigma \in \hat{\sigma}} \mathbf{F}_{\sigma}. \quad (5.1.1)$$

La base des \mathbf{P} est la *base fondamentale* de $M\mathbf{FQSym}$, l'algèbre de Hopf combinatoire ainsi définie. Afin d'être exhaustif et en but de référence future, énonçons le théorème suivant de [HN07] ainsi que sa démonstration.

Théorème 5.1.1. *Les éléments $\mathbf{P}_{\hat{\sigma}}$ définis en (5.1.1) engendrent une sous-algèbre de Hopf de **FQSym**.*

Démonstration. Montrons que le produit de deux éléments $\mathbf{P}_{\hat{\sigma}}$ et $\mathbf{P}_{\hat{\nu}}$ exprimé dans **FQSym** peut se réécrire dans la base des \mathbf{P} . Nous avons

$$\mathbf{P}_{\hat{\sigma}} \cdot \mathbf{P}_{\hat{\nu}} = \left(\sum_{\sigma \in \hat{\sigma}} \mathbf{F}_{\sigma} \right) \cdot \left(\sum_{\nu \in \hat{\nu}} \mathbf{F}_{\nu} \right) \quad (5.1.2)$$

$$= \sum_{\substack{\sigma \in \hat{\sigma} \\ \nu \in \hat{\nu}}} \mathbf{F}_{\sigma} \cdot \mathbf{F}_{\nu} \quad (5.1.3)$$

$$= \sum_{\substack{\sigma \in \hat{\sigma} \\ \nu \in \hat{\nu}}} \sum_{\pi \in \sigma \sqcup \nu} \mathbf{F}_{\pi}. \quad (5.1.4)$$

Soit π une permutation et supposons que l'élément \mathbf{F}_{π} apparaît dans (5.1.4). Pour réécrire cette expression dans la base des \mathbf{P} , il est nécessaire que pour toute permutation $\pi' \equiv \pi$, l'élément $\mathbf{F}_{\pi'}$ y apparaisse également. Soit (σ, ν) l'unique couple de permutations tel que $\pi \in \sigma \sqcup \nu$. Notons n la taille de σ et m la taille de ν . Comme \equiv est compatible aux restrictions aux intervalles d'alphabet,

$$\sigma = \pi|_{[1,n]} \equiv \pi'|_{[1,n]} =: \sigma', \quad (5.1.5)$$

et

$$\pi|_{[n+1,n+m]} \equiv \pi'|_{[n+1,n+m]}. \quad (5.1.6)$$

De plus, comme \equiv est compatible avec la déstandardisation,

$$\nu = \text{std}(\pi|_{[n+1,n+m]}) \equiv \text{std}(\pi'|_{[n+1,n+m]}) =: \nu'. \quad (5.1.7)$$

Ainsi, étant donné que $\pi' \in \sigma' \sqcup \nu'$, l'élément $\mathbf{F}_{\pi'}$ apparaît bien dans (5.1.4).

Montrons à présent que le coproduit d'un élément $\mathbf{P}_{\hat{\pi}}$ exprimé dans **FQSym** peut se réécrire dans le carré tensoriel de la base des \mathbf{P} . Nous avons

$$\Delta(\mathbf{P}_{\hat{\pi}}) = \sum_{\pi \in \hat{\pi}} \Delta(\mathbf{F}_{\pi}) \quad (5.1.8)$$

$$= \sum_{\pi \in \hat{\pi}} \sum_{\pi=u \cdot v} \mathbf{F}_{\text{std}(u)} \otimes \mathbf{F}_{\text{std}(v)}. \quad (5.1.9)$$

Soient σ et ν deux permutations et supposons que l'élément $\mathbf{F}_{\sigma} \otimes \mathbf{F}_{\nu}$ apparaît dans 5.1.9. Pour réécrire cette expression dans le carré tensoriel de la base des \mathbf{P} , il est nécessaire que pour toutes permutations $\sigma' \equiv \sigma$ et $\nu' \equiv \nu$, l'élément $\mathbf{F}_{\sigma'} \otimes \mathbf{F}_{\nu'}$ y apparaisse également. Par définition du coproduit de **FQSym**, il existe une permutation π telle que $\pi = u \cdot v$ avec $\text{std}(u) = \sigma$ et $\text{std}(v) = \nu$. Comme u et v sont des facteurs de permutations, il existe des réarrangements u' et v' des lettres de u et de v tels que $\text{std}(u') = \sigma'$ et $\text{std}(v') = \nu'$. Comme \equiv est compatible avec

la déstandardisation et que $\text{ev}(u') = \text{ev}(u)$ et $\text{ev}(v') = \text{ev}(v)$, on a $u \equiv u'$ et $v \equiv v'$. De plus, comme \equiv est une congruence, $u \cdot v \equiv u' \cdot v' =: \pi'$. Par conséquent, π' est dans la classe de π et l'élément $\mathbf{F}_{\sigma'} \otimes \mathbf{F}_{\nu'}$ apparaît dans (5.1.4).

Nous avons ainsi montré que le produit et le coproduit sont bien définis sur la base des \mathbf{P} de \mathbf{MFQSym} . Ils engendrent donc une sous-algèbre de Hopf de \mathbf{FQSym} . \square

Une classe d'équivalence de permutations pour la relation \equiv peut être représentée par sa permutation minimale pour l'ordre lexicographique. De ce fait, le produit et le coproduit de \mathbf{MFQSym} vérifient

$$\mathbf{P}_{\widehat{\sigma}} \cdot \mathbf{P}_{\widehat{\nu}} = \sum_{\substack{\sigma \in \widehat{\sigma} \\ \nu \in \widehat{\nu}}} \sum_{\substack{\pi \in \sigma \sqcup \nu \\ \pi = \min_{\leq_{\text{lex}}} \pi}} \mathbf{P}_{\widehat{\pi}}, \quad (5.1.10)$$

et

$$\Delta(\mathbf{P}_{\widehat{\pi}}) = \sum_{\pi \in \widehat{\pi}} \sum_{\substack{\pi = u \cdot v \\ u = \min_{\leq_{\text{lex}}} u \\ v = \min_{\leq_{\text{lex}}} v}} \mathbf{P}_{\widehat{\text{std}(u)}} \otimes \mathbf{P}_{\widehat{\text{std}(v)}}. \quad (5.1.11)$$

La compatibilité avec la déstandardisation de \equiv implique que toute classe d'équivalence de \equiv est constituée de mots de même longueur. Par conséquent, \mathbf{MFQSym} hérite de la graduation de \mathbf{FQSym} .

L'algèbre de Hopf \mathbf{MFQSym}^*

Regardons maintenant l'algèbre de Hopf duale \mathbf{MFQSym}^* . En suivant [DHNT08], les polynômes réalisateurs de \mathbf{MFQSym}^* sont les images des polynômes réalisateurs de \mathbf{FQSym}^* par la projection canonique

$$\tau : \mathbb{K}\langle A \rangle \rightarrow \mathbb{K}[A^*/\equiv] \simeq \mathbb{K}\langle A \rangle / \equiv, \quad (5.1.12)$$

où $\mathbb{K}[A^*/\equiv]$ est l'algèbre du monoïde M . Une autre description de \mathbf{MFQSym}^* parfaitement équivalente est la suivante. Soit I le sous-espace vectoriel de \mathbf{FQSym}^* défini par

$$I := \text{Vect}(\{\mathbf{F}_{\sigma}^* - \mathbf{F}_{\nu}^* : \sigma \equiv \nu \in \mathfrak{S}\}). \quad (5.1.13)$$

Les compatibilités de \equiv aux restrictions aux intervalles d'alphabet et à la déstandardisation impliquent que I est un idéal de Hopf de \mathbf{FQSym}^* . L'algèbre de Hopf \mathbf{MFQSym}^* admet alors l'expression

$$\mathbf{MFQSym}^* = \mathbf{FQSym}^* / I. \quad (5.1.14)$$

Notons $\tau : \mathbf{FQSym}^* \rightarrow \mathbf{MFQSym}^*$ la projection canonique définie linéairement par $\tau(\mathbf{F}_{\sigma}^*) := \mathbf{P}_{\widehat{\sigma}}^*$.

Le produit de \mathbf{MFQSym}^* vérifie

$$\mathbf{P}_{\widehat{\sigma}}^* \cdot \mathbf{P}_{\widehat{\nu}}^* = \tau(\mathbf{F}_{\sigma}^* \cdot \mathbf{F}_{\nu}^*), \quad (5.1.15)$$

où σ et ν sont deux permutations telles que $\sigma \in \widehat{\sigma}$ et $\nu \in \widehat{\nu}$. En se basant sur l'expression du produit de \mathbf{FQSym}^* donnée en (2.2.29) dans le chapitre 2, nous obtenons la formulation suivante :

$$\mathbf{P}_{\widehat{\sigma}}^* \cdot \mathbf{P}_{\widehat{\nu}}^* = \sum_{\substack{u \cdot v \in \mathfrak{S} \\ \text{std}(u) = \sigma \\ \text{std}(v) = \nu}} \mathbf{P}_{\widehat{u \cdot v}}^*. \quad (5.1.16)$$

Le coproduit de \mathbf{MFQSym}^* vérifie quant à lui

$$\Delta(\mathbf{P}_{\widehat{\pi}}^*) = (\tau \otimes \tau)(\Delta(\mathbf{F}_{\pi}^*)), \quad (5.1.17)$$

où π est une permutation telle que $\pi \in \hat{\pi}$. En se basant sur l'expression du coproduit de **FQSym**^{*} donnée en (2.2.30) dans le chapitre 2, nous obtenons la formulation suivante :

$$\Delta(\mathbf{P}_{\pi}^{\star}) = \sum_{\substack{0 \leq i \leq |\pi| \\ \sigma := \pi|_{[1, i]} \\ \nu := \text{std}(\pi|_{[i+1, |\pi|])}} \mathbf{P}_{\sigma}^{\star} \otimes \mathbf{P}_{\nu}^{\star}. \quad (5.1.18)$$

Donnons à présent trois exemples bien connus d'algèbres de Hopf obtenues selon cette construction.

5.1.2 L'algèbre de Hopf des tableaux de Young standard

Lorsque M est le monoïde plaxique (voir le paragraphe 4.1.2 du chapitre 4), l'algèbre de Hopf **MFQSym** est l'algèbre de Hopf de Poirier et Reutenauer des tableaux de Young standard [PR95]. Nous appelons cette structure **FSym** [DHT02] et la réalisation polynomiale de ses éléments, obtenue à partir de celle de **FQSym** rappelée en (2.2.37) dans le chapitre 2 forme l'algèbre des fonctions symétrique libres.

En notant $\mathbb{P}(u)$ et $\mathbb{Q}(u)$ les \mathbb{P} -symboles et \mathbb{Q} -symboles du mot u de la correspondance de Robinson-Schensted classique [Lot02], la réalisation polynomiale des éléments de la base fondamentale de **FSym** vérifie (voir [DHT02] et [Hiv03])

$$\mathbf{P}_T = \sum_{\substack{\sigma \in \mathfrak{S} \\ \mathbb{P}(\sigma) = T}} \mathbf{F}_{\sigma} = \sum_{\substack{u \in A^* \\ \mathbb{Q}(u) = T}} u, \quad (5.1.19)$$

pour tout tableau de Young standard T . Cette réalisation polynomiale fournit l'une des démonstrations les plus limpides [Lot02] de la règle de Littlewood-Richardson [LR34] qui décrit un moyen de calculer le produit de deux fonctions de Schur [Mac95]. Il est en effet possible, à partir du produit de deux éléments \mathbf{P}_T et $\mathbf{P}_{T'}$ de **FSym**, de calculer le produit des deux fonctions de Schur s_{λ} et $s_{\lambda'}$, où λ et λ' sont les partitions d'entiers de formes T et T' . Un fait remarquable est que la règle de Littlewood-Richardson fut découverte dans les années 1930, mais que les premières démonstrations furent trouvées seulement à partir des années 1970. La démonstration simple fournie en considérant le produit dans **FSym** illustre un point fort de cette théorie des sous-algèbres de Hopf de **FQSym** construites à partir de bons monoïdes.

L'algèbre de Hopf **FSym** possède par ailleurs quelques propriétés intéressantes. À titre d'exemple, les éléments de la base fondamentale qui apparaissent dans un produit impliquant deux éléments de la base fondamentale forment un intervalle d'un ordre partiel sur les tableaux de Young [Tas06]. Notons pour terminer que **FSym** n'est pas autoduale [DHNT08].

5.1.3 L'algèbre de Hopf des arbres binaires

L'algèbre de Hopf des arbres binaires fut découverte et construite originellement par Loday et Ronco [LR98]. Hivert, Novelli et Thibon l'ont construite à nouveau [HNT02] en passant par l'introduction du monoïde sylvestre (voir le paragraphe 4.1.2 du chapitre 4). Nous notons **PBT** l'algèbre de Hopf **MFQSym** où M est le monoïde sylvestre.

Nous donnons ici quelques exemples dans le but d'illustrer les concepts présentés dans le paragraphe 5.1.1. Tout d'abord, la base fondamentale de **PBT** est formée des éléments

$$\mathbf{P}_T := \sum_{\substack{\sigma \in \mathfrak{S} \\ \mathbb{P}(\sigma) = T}} \mathbf{F}_{\sigma}, \quad (5.1.20)$$

Le calcul du coproduit dans **PBT**^{*} est similaire : on calcule le coproduit de l'élément \mathbf{F}_σ^* dans **FQSym**^{*} où σ est une permutation arbitraire dans la classe sylvestre encodée par l'arbre binaire dont il est question, et on applique enfin τ sur chacun des éléments des tenseurs obtenus.

L'algèbre de Hopf **PBT** possède un certain nombre de propriétés qui la rendent particulièrement intéressante. Les éléments de sa base fondamentale qui apparaissent dans un produit impliquant deux éléments de sa base fondamentale forment un intervalle du treillis de Tamari (voir [HNT02] et [LR02]). De plus **PBT** est autoduale et l'isomorphisme de Hopf entre **PBT** et **PBT**^{*} est $\tau \circ \psi \circ \theta$ où θ est l'injection de **PBT** dans **FQSym** définie en suivant (5.1.20), et ψ est l'isomorphisme entre **FQSym** et **FQSym**^{*} défini par $\psi(\mathbf{F}_\sigma) := \mathbf{F}_{\sigma^{-1}}^*$.

5.1.4 L'algèbre de Hopf des fonctions symétriques non commutatives

Ce troisième exemple de sous-algèbre de Hopf de **FQSym** est classique. En choisissant pour M le monoïde hypoplaxique (voir le paragraphe 4.1.2 du chapitre 4), l'algèbre de Hopf **MFQSym** est l'algèbre de Hopf des fonctions symétriques non commutatives que nous notons **Sym**. Cette structure a été étudiée dans de nombreux travaux, comme par exemple [GKL⁺95], [KLT97], et [KT97]. Le dual de **Sym**, noté la plupart du temps **QSym** est l'algèbre de Hopf des fonctions quasi-symétriques [Ges84]. Notons que **Sym** n'est pas autoduale puisque **QSym** est commutative alors que **Sym** n'est pas.

Comme nous l'avons mentionné dans le paragraphe 4.1.2 du chapitre 4, les classes hypoplaxiques de permutations de taille n sont en bijection avec l'ensemble des mots binaires de longueur $n - 1$. Ainsi, la base fondamentale de **Sym** est indexée par l'ensemble des mots binaires. De plus, avec cet encodage, il est clair que le produit dans **Sym** vérifie

$$\mathbf{P}_b \cdot \mathbf{P}_{b'} = \mathbf{P}_{b \cdot 0 \cdot b'} + \mathbf{P}_{b \cdot 1 \cdot b'}. \quad (5.1.33)$$

Il est immédiat, sur les définitions des relations d'adjacence sylvestre et hypoplaxique, que toute classe d'équivalence hypoplaxique est une union de classes sylvestres. Cette remarque mène à l'existence du morphisme de Hopf injectif

$$\beta : \mathbf{Sym} \hookrightarrow \mathbf{PBT}, \quad (5.1.34)$$

vérifiant, en utilisant le lemme 4.3.3 du chapitre 4,

$$\beta(\mathbf{P}_b) = \sum_{\substack{T \in \mathcal{T} \\ \text{cnp}(T)=b}} \mathbf{P}_T, \quad (5.1.35)$$

pour tout mot binaire b . D'un point de vue combinatoire, β associe à un mot binaire b la somme des arbres binaires ayant b comme canopée. Cette application a été décrite par Loday et Ronco [LR98].

5.2 Quelques propriétés générales des sous-algèbres de Hopf de **FQSym**

Dans ce paragraphe, \equiv désigne une relation d'équivalence sur A^* qui est également une congruence de monoïde et compatible avec la déstandardisation et aux restrictions aux intervalles d'alphabet. Soit $M := A^*/\equiv$ le bon monoïde associé à \equiv et **MFQSym** la sous-algèbre de Hopf de **FQSym** associée à M obtenue selon la construction décrite dans le paragraphe 5.1.1.

5.2.1 Produits et intervalles

Rappelons que dans le cas où \equiv est une congruence du permutoèdre, la structure \mathfrak{S}_n/\equiv est pour tout $n \geq 0$ un treillis, et à plus forte raison, un poset. Cette construction est détaillée dans le paragraphe 4.4.1 du chapitre 4. Nous notons \leq_P la relation d'ordre partiel ainsi obtenue sur les classes de permutations de taille n pour la relation \equiv .

Lorsque \equiv est une congruence du permutoèdre, il se passe un phénomène particulier au niveau du produit de $MFQSym$. En effet, les éléments de la base des \mathbf{P} qui apparaissent dans un produit de deux éléments de la base des \mathbf{P} de $MFQSym$ forment un intervalle du treillis quotient \mathfrak{S}_n/\equiv . Avant de montrer cette propriété, nous avons besoin du lemme suivant.

Lemme 5.2.1. *Soient $\sigma^{(1)}$, $\sigma^{(2)}$, $\nu^{(1)}$ et $\nu^{(2)}$ des permutations qui vérifient $\sigma^{(1)} \leq_P \sigma^{(2)}$ et $\nu^{(1)} \leq_P \nu^{(2)}$. Nous avons alors dans $FQSym$,*

$$\left(\sum_{\sigma^{(1)} \leq_P \sigma \leq_P \sigma^{(2)}} \mathbf{F}_\sigma \right) \cdot \left(\sum_{\nu^{(1)} \leq_P \nu \leq_P \nu^{(2)}} \mathbf{F}_\nu \right) = \sum_{\sigma^{(1)} / \nu^{(1)} \leq_P \pi \leq_P \sigma^{(2)} \setminus \nu^{(2)}} \mathbf{F}_\pi. \quad (5.2.1)$$

Démonstration. Il est tout d'abord clair que tout élément \mathbf{F}_π qui apparaît dans le membre droit de (5.2.1) possède une multiplicité de 1. De plus, on déduit directement à partir de la définition du produit de mélange décalé sur les permutations qu'il en est de même pour le membre gauche de (5.2.1).

Supposons maintenant qu'un élément \mathbf{F}_π apparaît dans le membre gauche de (5.2.1). Il existe alors un unique couple (σ, ν) de permutations tel que $\pi \in \sigma \sqcup \nu$, $\sigma^{(1)} \leq_P \sigma \leq_P \sigma^{(2)}$ et $\nu^{(1)} \leq_P \nu \leq_P \nu^{(2)}$. Par définition du produit de mélange décalé, on a $\sigma / \nu \leq_P \pi \leq_P \sigma \setminus \nu$. Ceci implique $\sigma^{(1)} / \nu^{(1)} \leq_P \pi \leq_P \sigma^{(2)} \setminus \nu^{(2)}$, et par conséquent, \mathbf{F}_π apparaît dans le membre droit de (5.2.1).

Supposons qu'un élément \mathbf{F}_π apparaît dans le membre droit de (5.2.1). Nous avons alors $\sigma^{(1)} / \nu^{(1)} \leq_P \pi \leq_P \sigma^{(2)} \setminus \nu^{(2)}$. Posons $n := |\sigma^{(1)}| = |\sigma^{(2)}|$, $m := |\nu^{(1)}| = |\nu^{(2)}|$, $\sigma := \pi|_{[1,n]}$ et $\nu := \text{std}(\pi|_{[n+1, n+m]})$. Par définition du produit de mélange décalé, on a $\pi \in \sigma \sqcup \nu$ ainsi que $\sigma^{(1)} \leq_P \sigma \leq_P \sigma^{(2)}$ et $\nu^{(1)} \leq_P \nu \leq_P \nu^{(2)}$. L'élément \mathbf{F}_π apparaît donc également dans le membre gauche de (5.2.1). \square

Introduisons les notations suivantes. Si $\hat{\sigma}$ est une classe d'équivalence de permutations de \equiv , on note par $\hat{\sigma}\uparrow$ (resp. $\hat{\sigma}\downarrow$) l'élément minimal (resp. maximal) de la classe $\hat{\sigma}$ pour l'ordre du permutoèdre. On définit également pour tout $\sigma \in \hat{\sigma}$, les éléments $\sigma\uparrow := \hat{\sigma}\uparrow$ et $\sigma\downarrow := \hat{\sigma}\downarrow$. De plus, si $\hat{\sigma}$ et $\hat{\nu}$ sont des classes d'équivalence de permutations de \equiv , on note $\hat{\sigma} / \hat{\nu}$ la classe d'équivalence de la permutation $\hat{\sigma}\uparrow / \hat{\nu}\uparrow$. De même, on note $\hat{\sigma} \setminus \hat{\nu}$ la classe d'équivalence de la permutation $\hat{\sigma}\downarrow \setminus \hat{\nu}\downarrow$.

Proposition 5.2.2. *Soit $M := A^*/\equiv$ un bon monoïde où \equiv est une congruence du permutoèdre. Alors, pour toutes classes d'équivalence $\hat{\sigma}$ et $\hat{\nu}$ de \equiv de permutations, nous avons, dans $MFQSym$,*

$$\mathbf{P}_{\hat{\sigma}} \cdot \mathbf{P}_{\hat{\nu}} = \sum_{\hat{\sigma} / \hat{\nu} \leq \pi \leq \hat{\sigma} \setminus \hat{\nu}} \mathbf{P}_{\pi}. \quad (5.2.2)$$

Démonstration. Comme \equiv est une congruence du permutoèdre, les classes de \equiv sont des intervalles du permutoèdre, et ainsi $\hat{\sigma} = [\hat{\sigma}\uparrow, \hat{\sigma}\downarrow]$ et $\hat{\nu} = [\hat{\nu}\uparrow, \hat{\nu}\downarrow]$. De ce fait, et d'après le lemme 5.2.1, on a

$$\mathbf{P}_{\hat{\sigma}} \cdot \mathbf{P}_{\hat{\nu}} = \left(\sum_{\hat{\sigma}\uparrow \leq_P \sigma \leq_P \hat{\sigma}\downarrow} \mathbf{F}_\sigma \right) \cdot \left(\sum_{\hat{\nu}\uparrow \leq_P \nu \leq_P \hat{\nu}\downarrow} \mathbf{F}_\nu \right) \quad (5.2.3)$$

$$= \sum_{\hat{\sigma}\uparrow / \hat{\nu}\uparrow \leq_P \pi \leq_P \hat{\sigma}\downarrow \setminus \hat{\nu}\downarrow} \mathbf{F}_\pi. \quad (5.2.4)$$

D'après le théorème 5.1.1, l'expression (5.2.4) peut s'écrire comme une somme d'éléments de base **P**. Ceci, et la définition de la relation d'ordre \leq sur les classes de \equiv impliquent le résultat. \square

5.2.2 Bases multiplicatives et générateurs libres

Lorsque \equiv est une congruence du permutoèdre, les éléments de la base des **P** de **MFQSym** de chaque composante homogène sont munis de l'ordre partiel \leq . Cet ordre mène à la définition par triangularité de nouvelles bases pour **MFQSym**, analogues aux bases élémentaires et homogènes de **FQSym** (voir (2.2.32) et (2.2.33) dans le chapitre 2). Soit en effet la base des **E** définie par

$$\mathbf{E}_{\hat{\sigma}} := \sum_{\hat{\sigma} \leq \hat{\nu}} \mathbf{P}_{\hat{\nu}}, \quad (5.2.5)$$

que l'on appelle *base élémentaire* de **MFQSym**. On définit également la base des **H** par

$$\mathbf{H}_{\hat{\sigma}} := \sum_{\hat{\nu} \leq \hat{\sigma}} \mathbf{P}_{\hat{\nu}}, \quad (5.2.6)$$

que l'on appelle *base homogène* de **MFQSym**.

Lemme 5.2.3. *Soit $M := A^*/\equiv$ un bon monoïde où \equiv est une congruence du permutoèdre. Alors, pour toute classe d'équivalence $\hat{\sigma}$ de \equiv de permutations, nous avons dans **MFQSym**,*

$$\mathbf{E}_{\hat{\sigma}} = \mathbf{E}^{\hat{\sigma}\uparrow}, \quad (5.2.7)$$

et

$$\mathbf{H}_{\hat{\sigma}} = \mathbf{H}^{\hat{\sigma}\downarrow}, \quad (5.2.8)$$

où les éléments $\mathbf{E}^{\hat{\sigma}\uparrow}$ et $\mathbf{H}^{\hat{\sigma}\downarrow}$ sont définis dans **FQSym**.

Démonstration. En utilisant le fait que \equiv est une congruence du permutoèdre, nous avons successivement

$$\mathbf{E}_{\hat{\sigma}} = \sum_{\hat{\sigma} \leq \hat{\nu}} \mathbf{P}_{\hat{\nu}} = \sum_{\hat{\sigma} \leq \hat{\nu}} \sum_{\nu \in \hat{\nu}} \mathbf{F}_{\nu} = \sum_{\hat{\sigma}\uparrow \leq_P \nu} \mathbf{F}_{\nu} = \mathbf{E}^{\hat{\sigma}\uparrow}. \quad (5.2.9)$$

La démonstration de (5.2.8) est analogue. \square

Proposition 5.2.4. *Soit $M := A^*/\equiv$ un bon monoïde où \equiv est une congruence du permutoèdre. Alors, pour toutes classes d'équivalence $\hat{\sigma}$ et $\hat{\nu}$ de \equiv de permutations, nous avons, dans **MFQSym**,*

$$\mathbf{E}_{\hat{\sigma}} \cdot \mathbf{E}_{\hat{\nu}} = \mathbf{E}_{\hat{\sigma} \wedge \hat{\nu}}, \quad (5.2.10)$$

et

$$\mathbf{H}_{\hat{\sigma}} \cdot \mathbf{H}_{\hat{\nu}} = \mathbf{H}_{\hat{\sigma} \vee \hat{\nu}}. \quad (5.2.11)$$

Démonstration. D'après le lemme 5.2.3 et l'expression du produit dans la base élémentaire de **FQSym** rappelée en (2.2.34) dans le chapitre 2, nous avons

$$\mathbf{E}_{\hat{\sigma}} \cdot \mathbf{E}_{\hat{\nu}} = \mathbf{E}^{\hat{\sigma}\uparrow} \cdot \mathbf{E}^{\hat{\nu}\uparrow} = \mathbf{E}^{\hat{\sigma}\uparrow \wedge \hat{\nu}\uparrow}. \quad (5.2.12)$$

Il reste maintenant à montrer que la permutation $\hat{\sigma}\uparrow \wedge \hat{\nu}\uparrow =: \pi$ est le plus petit élément de sa classe. La permutation π est de la forme $\pi = u \cdot v$ où u est de longueur $|\hat{\sigma}\uparrow|$ et v de longueur $|\hat{\nu}\uparrow|$. Comme toutes les lettres de u sont strictement inférieures à celles de v , s'il existe une permutation π' couverte par π dans l'ordre du permutoèdre, alors il existe nécessairement dans u ou dans v un facteur de la forme \mathbf{ba} avec $\mathbf{a} < \mathbf{b}$. Cependant, comme u et $\text{std}(v)$ sont déjà les plus petits éléments de leurs classes respectives, la permutation obtenue en remplaçant le facteur \mathbf{ba} de π par \mathbf{ab} n'est pas dans la classe de π . La permutation π est donc le plus petit élément de sa classe, ce qui implique que (5.2.12) est égal, d'après le lemme 5.2.3, à $\mathbf{E}_{\hat{\sigma} \wedge \hat{\nu}}$. La démonstration de la seconde partie de la proposition est analogue. \square

Une classe d'équivalence $\hat{\sigma}$ de \equiv de permutations est dite *connexe* si la permutation $\hat{\sigma}\uparrow$ est connexe. De même, nous dirons que $\hat{\sigma}$ est *anti-connexe* si la permutation $\hat{\sigma}\downarrow$ est anti-connexe. Remarquons que puisque pour toute permutation σ connexe (resp. anti-connexe) et une permutation ν telles que $\sigma \leq_P \nu$ (resp. $\nu \leq_P \sigma$), la permutation ν est également connexe (resp. anti-connexe), pour décider si une classe d'équivalence $\hat{\sigma}$ de \equiv de permutations est connexe (resp. anti-connexe), il suffit de tester si la permutation $\hat{\sigma}\uparrow$ (resp. $\hat{\sigma}\downarrow$) est connexe (resp. anti-connexe).

Proposition 5.2.5. *Soit $M := A^*/\equiv$ un bon monoïde où \equiv est une congruence du permutoèdre. Alors, l'algèbre $MFQSym$ est engendrée librement par les éléments $\mathbf{E}_{\hat{\sigma}}$ (resp. $\mathbf{H}_{\hat{\sigma}}$) tels que $\hat{\sigma}$ est connexe (resp. anti-connexe).*

Démonstration. Étant donné qu'il n'y a pas de relations entre les éléments \mathbf{E}^σ de $FQSym$ où σ est une permutation connexe, et que, d'après le lemme 5.2.3, $\mathbf{E}_{\hat{\sigma}} = \mathbf{E}^{\hat{\sigma}\uparrow}$ pour toute permutation σ , la sous-algèbre de $MFQSym$ engendrée par les $\mathbf{E}_{\hat{\sigma}}$ avec $\hat{\sigma}$ connexe est libre.

Montrons maintenant que $MFQSym$ est engendrée par les éléments $\mathbf{E}_{\hat{\sigma}}$ avec $\hat{\sigma}$ connexe. Soit $\hat{\sigma}$ une classe de \equiv de permutations. Il existe alors une expression de la forme

$$\hat{\sigma}\uparrow = \sigma^{(1)} / \dots / \sigma^{(\ell)}, \quad (5.2.13)$$

où les permutations $\sigma^{(1)}, \dots, \sigma^{(\ell)}$ sont connexes. Nous avons, par l'expression du produit dans la base élémentaire de $FQSym$, l'égalité

$$\mathbf{E}^{\hat{\sigma}\uparrow} = \mathbf{E}^{\sigma^{(1)}} \cdot \dots \cdot \mathbf{E}^{\sigma^{(\ell)}}. \quad (5.2.14)$$

Comme les permutations $\sigma^{(1)}, \dots, \sigma^{(\ell)}$ sont clairement les plus petits éléments de leurs classes respectives, d'après le lemme 5.2.3, nous avons dans $MFQSym$ l'égalité

$$\mathbf{E}_{\hat{\sigma}} = \mathbf{E}_{\sigma^{(1)}} \cdot \dots \cdot \mathbf{E}_{\sigma^{(\ell)}}, \quad (5.2.15)$$

ce qui montre le résultat. La démonstration de la seconde partie de la proposition est analogue. \square

5.2.3 Bases booléennes

Dorénavant, la relation d'équivalence \equiv n'est plus nécessairement une congruence du permutoèdre.

Une base \mathbf{B} d'une algèbre est une *base booléenne d'algèbre* si tout élément de la base \mathbf{B} qui apparaît dans un produit de deux éléments de base admet un coefficient à 1. De la même manière, une base \mathbf{B} d'une cogèbre est une *base booléenne de cogèbre* si tout élément du carré tensoriel de la base \mathbf{B} qui apparaît dans un coproduit d'un élément de base admet un coefficient à 1. Finalement, une base \mathbf{B} d'une algèbre de Hopf (H, \cdot, Δ) est une *base booléenne* si \mathbf{B} est à la fois une base booléenne d'algèbre de (H, \cdot) et une base booléenne de cogèbre de (H, Δ) .

Nous donnons ici un critère combinatoire suffisant sur M pour que la base fondamentale de $MFQSym$ soit booléenne.

Proposition 5.2.6. *Soit $M := A^*/\equiv$ un bon monoïde tel que, pour toutes permutations π et μ ,*

$$\sigma, \nu \in \pi \sqcup \mu \quad \text{et} \quad \sigma^{-1} \equiv \nu^{-1} \quad \text{implique} \quad \sigma = \nu. \quad (5.2.16)$$

Alors la base des \mathbf{P} est une base booléenne de $MFQSym$.

Démonstration. D'après l'expression du produit de $MFQSym$ donnée en (5.1.10), et par la définition du produit de mélange décalé sur les permutations, il est immédiat que \mathbf{P} est une base booléenne d'algèbre, et ce, indépendamment du respect de la condition (5.2.16).

Par dualité, la base des \mathbf{P} est une base booléenne de la cogèbre $M\mathbf{FQSym}$ si et seulement si sa base duale \mathbf{P}^* est une base booléenne de l'algèbre $M\mathbf{FQSym}$. Nous avons

$$\mathbf{P}_\pi^* \cdot \mathbf{P}_\mu^* = \tau(\mathbf{F}_\pi^* \cdot \mathbf{F}_\mu^*) \quad (5.2.17)$$

$$= \tau(\psi(\psi^{-1}(\mathbf{F}_\pi^*) \cdot \psi^{-1}(\mathbf{F}_\mu^*))) \quad (5.2.18)$$

$$= \tau(\psi(\mathbf{F}_{\pi^{-1}} \cdot \mathbf{F}_{\mu^{-1}})) \quad (5.2.19)$$

$$= \sum_{\sigma \in \pi^{-1} \sqcup \mu^{-1}} \tau(\mathbf{F}_{\sigma^{-1}}^*), \quad (5.2.20)$$

où $\pi \in \widehat{\pi}$, $\mu \in \widehat{\mu}$ et $\psi : \mathbf{FQSym} \rightarrow \mathbf{FQSym}^*$ est l'isomorphisme d'algèbres de Hopf défini par $\psi(\mathbf{F}_\sigma) := \mathbf{F}_{\sigma^{-1}}^*$. Nous observons que si \equiv vérifie l'hypothèse (5.2.16), alors il n'y a pas de multiplicités dans (5.2.20). \square

5.2.4 Structure de bigèbre bidendriforme

Algèbres dendriformes et structure dendriforme de **FQSym**

Commençons par rappeler la notion d'algèbre dendriforme due à Loday [Lod01] (voir par ailleurs [Lod02] et [Lod04]).

Définition 5.2.7. Une algèbre dendriforme est un espace vectoriel D muni de deux applications linéaires $\prec : D \otimes D \rightarrow D$ et $\succ : D \otimes D \rightarrow D$ appelées usuellement respectivement gauche et droite telles que, pour tous $x, y, z \in D$,

$$(x \prec y) \prec z = x \prec (y \prec z) + x \prec (y \succ z), \quad (5.2.21)$$

$$(x \succ y) \prec z = x \succ (y \prec z), \quad (5.2.22)$$

$$(x \prec y) \succ z + (x \succ y) \succ z = x \succ (y \succ z). \quad (5.2.23)$$

Toute algèbre sur l'opérade dendriforme $Dend$ est une algèbre dendriforme (voir le paragraphe 3.2.4 du chapitre 3). En outre, l'application linéaire $\prec + \succ$ d'une algèbre dendriforme (D, \prec, \succ) est associative. Toute algèbre dendriforme permet donc de définir une algèbre associative.

Cette remarque peut être prise dans l'autre sens et l'on peut s'intéresser à construire une algèbre dendriforme à partir d'une algèbre associative. Ainsi, une algèbre associative (A, \cdot) admet une structure dendriforme s'il existe une algèbre dendriforme (A, \prec, \succ) telle que $\cdot = \prec + \succ$ et \prec et \succ ne sont pas dégénérées, i.e., il existe $x, x', y, y' \in A$ tels que $x \prec y \neq 0$ et $x' \succ y' \neq 0$. Dans le cas où le produit \cdot admet une unité 1, on impose, pour tout $x \in A$ tel que $x \neq 1$, les relations suivantes :

$$1 \prec x = 0 = x \succ 1, \quad (5.2.24)$$

$$1 \succ x = x = x \prec 1, \quad (5.2.25)$$

et les expressions $1 \prec 1$ et $1 \succ 1$ ne sont pas définies.

Donnons à présent quelques exemples d'algèbres dendriformes.

Soit l'espace vectoriel $D := \mathbb{K}\langle A \rangle$ des polynômes sur A non commutatifs, que l'on munit des applications linéaires \prec et \succ définies linéairement pour tout $u, v \in A^+$ par

$$u \prec v := (u_1 \dots u_{|u|-1} \sqcup v) \cdot u_{|u|}, \quad (5.2.26)$$

$$u \succ v := (u \sqcup v_1 \dots v_{|v|-1}) \cdot v_{|v|}. \quad (5.2.27)$$

La structure (D, \prec, \succ) est l'*algèbre dendriforme de mélange*. L'algèbre associative obtenue en sommant \prec et \succ est (D, \sqcup) , l'algèbre de mélange que nous avons déjà rencontrée dans le paragraphe 2.2.1 du chapitre 2.

Sur ce même espace vectoriel D , on peut définir une autre structure d'algèbre dendriforme (D, \ll, \gg) où \ll et \gg sont définies linéairement pour tout $u, v \in A^+$ par

$$u \ll v := \begin{cases} u \cdot v & \text{si } \max(u) \geq \max(v), \\ 0 & \text{sinon,} \end{cases} \quad (5.2.28)$$

$$u \gg v := \begin{cases} u \cdot v & \text{si } \max(u) < \max(v), \\ 0 & \text{sinon.} \end{cases} \quad (5.2.29)$$

C'est l'*algèbre dendriforme max* [NT07]. L'algèbre associative obtenue à partir de cette dernière est l'algèbre des polynômes non commutatifs, munie de son produit usuel.

Les produits gauche et droite de l'*algèbre dendriforme libre* sur un générateur possèdent une description en termes de mélanges d'arbres binaires. De plus, l'algèbre dendriforme libre est, en tant qu'espace vectoriel, l'espace vectoriel combinatoire des arbres binaires [LR98]. D'autres descriptions de cette structure existent et ont été étudiées, notamment en termes de polynômes [Rey06] formant une sous-algèbre dendriforme de l'algèbre dendriforme max et en termes de forêts ordonnées d'arbres plans enracinés [Foi09]. Un fait remarquable est que l'algèbre associative obtenue depuis l'algèbre dendriforme libre sur un générateur est isomorphe à **PBT** [HNT05].

Notre dernier exemple d'algèbre dendriforme est du à Loday et Ronco [LR02]. Munissons l'espace vectoriel **FQSym** des deux applications linéaires \prec et \succ définies linéairement pour tous $\sigma, \nu \in \mathfrak{S}_n$ où $n \geq 1$ par

$$\mathbf{F}_\sigma \prec \mathbf{F}_\nu := \sum_{\substack{\pi \in \sigma \sqcup \nu \\ \pi|_\pi = \sigma|_\sigma}} \mathbf{F}_\pi, \quad (5.2.30)$$

$$\mathbf{F}_\sigma \succ \mathbf{F}_\nu := \sum_{\substack{\pi \in \sigma \sqcup \nu \\ \pi|_\pi = \nu|_\nu + |\sigma|}} \mathbf{F}_\pi. \quad (5.2.31)$$

Voici quelques exemples :

$$\mathbf{F}_{12} \prec \mathbf{F}_{21} = \mathbf{F}_{1432} + \mathbf{F}_{4132} + \mathbf{F}_{4312}, \quad (5.2.32)$$

$$\mathbf{F}_{12} \succ \mathbf{F}_{21} = \mathbf{F}_{1243} + \mathbf{F}_{1423} + \mathbf{F}_{4123}, \quad (5.2.33)$$

$$\mathbf{F}_{21} \prec \mathbf{F}_{312} = \mathbf{F}_{25341} + \mathbf{F}_{52341} + \mathbf{F}_{53241} + \mathbf{F}_{53421}, \quad (5.2.34)$$

$$\mathbf{F}_{21} \succ \mathbf{F}_{312} = \mathbf{F}_{21534} + \mathbf{F}_{25134} + \mathbf{F}_{25314} + \mathbf{F}_{52134} + \mathbf{F}_{52314} + \mathbf{F}_{53214}. \quad (5.2.35)$$

La structure $(\mathbf{FQSym}^+, \prec, \succ)$, où \mathbf{FQSym}^+ est l'espace vectoriel **FQSym** privé de sa composante homogène de degré 0, est une algèbre dendriforme, et de plus, l'algèbre **FQSym** avec son produit de mélange décalé est précisément l'algèbre associative obtenue en sommant les produits \prec et \succ et en ajoutant l'unité \mathbf{F}_ϵ .

Cogèbres codendriformes et structure codendriforme de **FQSym**

En dualisant les applications linéaires \prec et \succ de la définition 5.2.7, on obtient un autre type de structure algébrique. Plus précisément :

Définition 5.2.8. Une cogèbre codendriforme est un espace vectoriel C muni de deux applications linéaires $\Delta_{\prec} : C \rightarrow C \otimes C$ et $\Delta_{\succ} : C \rightarrow C \otimes C$ telles que, pour tout $x \in C$,

$$(\Delta_{\prec} \otimes I) \circ \Delta_{\prec}(x) = (I \otimes \Delta_{\prec}) \circ \Delta_{\prec}(x) + (I \otimes \Delta_{\succ}) \circ \Delta_{\prec}(x), \quad (5.2.36)$$

$$(\Delta_{\succ} \otimes I) \circ \Delta_{\prec}(x) = (I \otimes \Delta_{\prec}) \circ \Delta_{\succ}(x), \quad (5.2.37)$$

$$(\Delta_{\prec} \otimes I) \circ \Delta_{\succ}(x) + (\Delta_{\succ} \otimes I) \circ \Delta_{\succ}(x) = (I \otimes \Delta_{\succ}) \circ \Delta_{\succ}(x), \quad (5.2.38)$$

où $I : C \rightarrow C$ est l'application identité.

Dans ce cas ci encore, l'application $\Delta_{\prec} + \Delta_{\succ}$ d'une cogèbre codendriforme $(C, \Delta_{\prec}, \Delta_{\succ})$ est associative et mène à la construction d'une cogèbre.

Suivant la terminologie posée plus haut, nous dirons qu'une cogèbre (C, Δ) admet une structure codendriforme s'il existe une cogèbre codendriforme $(C, \Delta_{\prec}, \Delta_{\succ})$ telle que $\Delta = \Delta_{\prec} + \Delta_{\succ}$, et Δ_{\prec} et Δ_{\succ} ne sont pas dégénérées, *i.e.*, il existe $x, x' \in C$ tels que $\Delta_{\prec}(x) \neq 0$ et $\Delta_{\succ}(x') \neq 0$. Dans le cas où le coproduit Δ admet une counité 1, on demande à avoir

$$\Delta = 1 \otimes I + \Delta_{\prec} + \Delta_{\succ} + I \otimes 1, \quad (5.2.39)$$

où $I : C \rightarrow C$ est l'application identité. De plus, les expressions $\Delta_{\prec}(1)$ et $\Delta_{\succ}(1)$ ne sont pas définies.

Foissy a introduit dans [Foi07] les applications linéaires Δ_{\prec} et Δ_{\succ} sur \mathbf{FQSym}^+ . Ces dernières sont définies linéairement pour tout $\sigma \in \mathfrak{S}_n$ avec $n \geq 1$ par

$$\Delta_{\prec}(\mathbf{F}_{\sigma}) := \sum_{\substack{\sigma = u \cdot v, |v| \geq 1 \\ \max(u) = \max(\sigma)}} \mathbf{F}_{\text{std}(u)} \otimes \mathbf{F}_{\text{std}(v)}, \quad (5.2.40)$$

$$\Delta_{\succ}(\mathbf{F}_{\sigma}) := \sum_{\substack{\sigma = u \cdot v, |u| \geq 1 \\ \max(v) = \max(\sigma)}} \mathbf{F}_{\text{std}(u)} \otimes \mathbf{F}_{\text{std}(v)}. \quad (5.2.41)$$

Voici quelques exemples :

$$\Delta_{\prec}(\mathbf{F}_{123}) = 0, \quad (5.2.42)$$

$$\Delta_{\succ}(\mathbf{F}_{123}) = \mathbf{F}_1 \otimes \mathbf{F}_{12} + \mathbf{F}_{12} \otimes \mathbf{F}_1, \quad (5.2.43)$$

$$\Delta_{\prec}(\mathbf{F}_{2413}) = \mathbf{F}_{12} \otimes \mathbf{F}_{12} + \mathbf{F}_{231} \otimes \mathbf{F}_1, \quad (5.2.44)$$

$$\Delta_{\succ}(\mathbf{F}_{2413}) = \mathbf{F}_1 \otimes \mathbf{F}_{312}. \quad (5.2.45)$$

La structure $(\mathbf{FQSym}^+, \Delta_{\prec}, \Delta_{\succ})$ est une cogèbre codendriforme. De plus, la cogèbre **FQSym** avec la counité \mathbf{F}_{ϵ} et son coproduit de déconcaténation standardisée est précisément la cogèbre obtenue à partir de cette cogèbre codendriforme.

Bigèbres bidendriformes et structure bidendriforme de **FQSym**

La notion de *bigèbre bidendriforme*, due à Foissy [Foi07], [Foi09] est aux algèbres dendriformes et cogèbres codendriformes ce que la notion d'algèbre de Hopf est aux algèbres et aux cogèbres. Plus exactement,

Définition 5.2.9. Une bigèbre bidendriforme est un espace vectoriel B muni de quatre applications linéaires $\prec : B \otimes B \rightarrow B$, $\succ : B \otimes B \rightarrow B$, $\Delta_{\prec} : B \rightarrow B \otimes B$ et $\Delta_{\succ} : B \rightarrow B \otimes B$ telles que

(i) (B, \prec, \succ) est une algèbre dendriforme ;

(ii) $(B, \Delta_{\prec}, \Delta_{\succ})$ est une cogèbre codendriforme ;

(iii) soient $\cdot := \prec + \succ$, $\Delta := \Delta_{\prec} + \Delta_{\succ}$, et en utilisant la notation de Sweedler, $\Delta(x) =: \sum x^L \otimes x^R$, $\Delta_{\prec}(x) =: \sum x_{\prec}^L \otimes x_{\prec}^R$, $\Delta_{\succ}(x) =: \sum x_{\succ}^L \otimes x_{\succ}^R$. Les éléments $x, y \in B$ vérifient

$$\begin{aligned} \Delta_{\succ}(x \succ y) = & \sum x^L \cdot y_{\succ}^L \otimes x^R \succ y_{\succ}^R + x^L \otimes x^R \succ y + y_{\succ}^L \otimes x \succ y_{\succ}^R \\ & + x \cdot y_{\succ}^L \otimes y_{\succ}^R + x \otimes y, \end{aligned} \quad (5.2.46)$$

$$\Delta_{\succ}(x \prec y) = \sum x_L \cdot y_{\succ}^L \otimes x^R \prec y_{\succ}^R + x^L \otimes x^R \prec y + y_{\succ}^L \otimes x \prec y_{\succ}^R, \quad (5.2.47)$$

$$\Delta_{\prec}(x \succ y) = \sum x^L \cdot y_{\prec}^L \otimes x^R \succ y_{\prec}^R + x \cdot y_{\prec}^L \otimes y_{\prec}^R + y_{\prec}^L \otimes x \succ y_{\prec}^R, \quad (5.2.48)$$

$$\begin{aligned} \Delta_{\prec}(x \prec y) = & \sum x^L \cdot y_{\prec}^L \otimes x^R \prec y_{\prec}^R + x^L \cdot y \otimes x^R + y_{\prec}^L \otimes x \prec y_{\prec}^R \\ & + y \otimes x. \end{aligned} \quad (5.2.49)$$

Nous dirons qu'une algèbre de Hopf (H, \cdot, Δ) admet une structure bidendriforme si l'algèbre (H, \cdot) admet une structure dendriforme (H, \prec, \succ) et la cogèbre $(H, \Delta_{\prec}, \Delta_{\succ})$ admet une structure codendriforme $(H, \Delta_{\prec}, \Delta_{\succ})$ telles que la structure $(H^+, \prec, \succ, \Delta_{\prec}, \Delta_{\succ})$ soit une bigèbre bidendriforme, où H^+ est l'idéal d'augmentation de H , i.e., le noyau de la counité de H .

Un élément x d'une bigèbre bidendriforme est un élément totalement primitif si

$$\Delta_{\prec}(x) = 0 = \Delta_{\succ}(x). \quad (5.2.50)$$

Notre intérêt pour cette structure algébrique est révélée par le théorème suivant de Foissy établi en [Foi07], cité ici pour référence future.

Théorème 5.2.10. *Si H est une algèbre de Hopf qui admet une structure bidendriforme, alors, H est autoduale, libre et colibre. De plus, l'algèbre de Lie engendrée par ses éléments primitifs est libre et l'algèbre dendriforme engendrée par ses éléments totalement primitifs l'est également.*

Foissy a aussi montré [Foi07] que **FQSym** admet une structure bidendriforme avec les produits (5.2.30) et (5.2.31), et les coproduits (5.2.40) et (5.2.41). Par conséquent, **FQSym** possède l'ensemble des propriétés énoncées dans le théorème 5.2.10.

Nous pouvons maintenant établir le point principal de ce paragraphe :

Proposition 5.2.11. *Soit $M := A^*/\equiv$ un bon monoïde tel que pour tous mots u et v de longueur $n \geq 1$,*

$$u \equiv v \quad \text{implique} \quad u_n = v_n. \quad (5.2.51)$$

Alors, MFQSym admet une structure bidendriforme. De plus, MFQSym est autoduale, libre, colibre, l'algèbre de Lie engendrée par ses éléments primitifs est libre, et l'algèbre dendriforme engendrée par ses éléments totalement primitifs l'est également.

Démonstration. Il suffit de montrer que les produits \prec et \succ (voir (5.2.30) et (5.2.31)), et les coproduits Δ_{\prec} et Δ_{\succ} (voir (5.2.40) et (5.2.41)) définis dans **FQSym** restent bien définis dans **MFQSym**. On montre ainsi de cette manière que **MFQSym** admet une structure bidendriforme et le reste de la proposition est impliqué par le théorème 5.2.11.

Soient $\hat{\sigma}$ et $\hat{\nu}$ deux classes d'équivalence de \equiv de permutations et \mathbf{F}_{π} un élément qui apparaît dans le produit $\mathbf{P}_{\hat{\sigma}}^{\wedge} \prec \mathbf{P}_{\hat{\nu}}^{\wedge}$. Il existe alors une permutation $\sigma \in \hat{\sigma}$ telle que $\pi_{|\sigma|} = \sigma_{|\sigma|}$. Soit π' une permutation telle que $\pi \equiv \pi'$. D'après le théorème 5.1.1, l'élément $\mathbf{F}_{\pi'}$ apparaît dans le produit $\mathbf{P}_{\hat{\sigma}}^{\wedge} \cdot \mathbf{P}_{\hat{\nu}}^{\wedge}$, et par conséquent, apparaît aussi dans $\mathbf{P}_{\hat{\sigma}}^{\wedge} \prec \mathbf{P}_{\hat{\nu}}^{\wedge}$ ou dans $\mathbf{P}_{\hat{\sigma}}^{\wedge} \succ \mathbf{P}_{\hat{\nu}}^{\wedge}$. Supposons par

l'absurde que $\mathbf{F}_{\pi'}$ apparaît dans $\mathbf{P}_{\hat{\sigma}} \succ \mathbf{P}_{\hat{\nu}}$. Il existe alors deux permutations $\sigma' \in \hat{\sigma}$ et $\nu' \in \hat{\nu}$ telles que $\pi'_{|\pi'|} = \nu'_{|\nu'|} + |\sigma'|$. Ceci implique $\pi_{|\pi|} \neq \pi'_{|\pi'|}$ et contredit le fait que tous les mots d'une même classe d'équivalence de \equiv se terminent par la même lettre. L'élément $\mathbf{F}_{\pi'}$ apparaît donc dans $\mathbf{P}_{\hat{\sigma}} \prec \mathbf{P}_{\hat{\nu}}$, ce qui montre que le produit \prec est bien défini dans \mathbf{MFQSym} . Il en est ainsi de même pour \succ car $\prec + \succ$ est le produit usuel de \mathbf{MFQSym} .

Soit $\hat{\pi}$ une classe d'équivalence de \equiv de permutations et un élément $\mathbf{F}_{\sigma} \otimes \mathbf{F}_{\nu}$ qui apparaît dans le coproduit $\Delta_{\prec}(\mathbf{P}_{\hat{\pi}})$. Il existe alors une permutation $\pi \in \hat{\pi}$ telle que $\pi = u \cdot v$, $\sigma = \text{std}(u)$, $\nu = \text{std}(v)$ et la plus grande lettre de $u \cdot v$ est dans le facteur u . Soient maintenant σ' et ν' deux permutations telles que $\sigma \equiv \sigma'$ et $\nu \equiv \nu'$. Montrons que l'élément $\mathbf{F}_{\sigma'} \otimes \mathbf{F}_{\nu'}$ apparaît dans le coproduit $\Delta_{\prec}(\mathbf{P}_{\hat{\pi}})$. Soient pour cela u' un réarrangement des lettres de u tel que $\text{std}(u') = \sigma'$ et v' un réarrangement des lettres de v tel que $\text{std}(v') = \nu'$. Comme $\text{ev}(u') = \text{ev}(u)$, $\text{std}(u') \equiv \text{std}(u)$ et que \equiv est compatible avec la déstandardisation, on a $u \equiv u'$. Pour la même raison, on a $v \equiv v'$, et, puisque \equiv est une congruence de monoïde, on a $u \cdot v \equiv u' \cdot v'$. Finalement, étant donné que la plus grande lettre de $u \cdot v$ se trouve dans u , la plus grande lettre de $u' \cdot v'$ se trouve dans u' . Ceci montre que l'élément $\mathbf{F}_{\sigma'} \otimes \mathbf{F}_{\nu'}$ apparaît dans $\Delta_{\prec}(\mathbf{P}_{\hat{\pi}})$. Le coproduit Δ_{\prec} est par conséquent bien défini dans \mathbf{MFQSym} . La démonstration est analogue pour le coproduit Δ_{\succ} . \square

5.3 L'algèbre de Hopf combinatoire des couples d'arbres binaires jumeaux

Nous utilisons dans ce paragraphe les propriétés du monoïde de Baxter énoncées dans le chapitre 4 pour construire la sous-algèbre de Hopf **Baxter** de \mathbf{FQSym} . Cette dernière est ensuite étudiée sous la lumière des concepts présentés dans le paragraphe 5.2.

Dans ce qui suit, tout couple d'arbres binaires jumeaux est quoiqu'il arrive — sauf mention explicite — étiqueté par une permutation de façon à ce qu'il soit un couple d'arbres binaires de recherche jumeaux. Rappelons qu'étant donné un couple d'arbres binaires jumeaux, il existe une unique façon de l'étiqueter de cette sorte. Ainsi, pour alléger nos représentations graphiques, les étiquettes des arbres ne sont pas dessinées.

5.3.1 L'algèbre de Hopf Baxter

Il a été montré dans le chapitre 4 que le monoïde de Baxter est un bon monoïde. Il est en effet, d'après la proposition 4.2.4 compatible avec la déstandardisation et d'après la proposition 4.2.5, compatible aux restrictions aux intervalles d'alphabet. De plus, d'après le théorème 4.3.12, toute classe d'équivalence de Baxter de permutations peut être encodée par un couple d'arbres binaires jumeaux et le \mathbb{P} -symbole de Baxter formulé dans la définition 4.3.1 envoie une permutation vers le couple d'arbres binaires jumeaux qui encode sa classe. De ce fait, en utilisant le théorème 5.1.1, nous obtenons le théorème suivant :

Théorème 5.3.1. *Les éléments \mathbf{P}_J , où J est un couple d'arbres binaires jumeaux, définis par*

$$\mathbf{P}_J := \sum_{\substack{\sigma \in \mathfrak{S} \\ \mathbb{P}(\sigma) = J}} \mathbf{F}_{\sigma}, \quad (5.3.1)$$

*engendrent une sous-algèbre de Hopf de \mathbf{FQSym} . Cette algèbre de Hopf est nommée **Baxter**.*

Voici à titre d'exemple quelques éléments de la base fondamentale de **Baxter** exprimée dans

Théorème 5.3.2. *L'application $r_A : \mathbf{Baxter} \rightarrow \mathbb{K}\langle A \rangle$ définie par*

$$r_A(\mathbf{P}_J) := \sum_{\substack{u \in A^* \\ (\text{croi}(u), \text{decr}(u)) \simeq J}} u, \quad (5.3.10)$$

*fournit une réalisation polynomiale de **Baxter**.*

Démonstration. Appliquons la réalisation polynomiale r_A de **FQSym** sur les éléments de la base fondamentale de **Baxter** :

$$r_A(\mathbf{P}_J) = \sum_{\substack{\sigma \in \mathfrak{S} \\ \mathbb{P}(\sigma) = J}} r_A(\mathbf{F}_\sigma) \quad (5.3.11)$$

$$= \sum_{\substack{\sigma \in \mathfrak{S} \\ \mathbb{P}(\sigma^{-1}) = J}} r_A(\mathbf{G}_\sigma) \quad (5.3.12)$$

$$= \sum_{\substack{\sigma \in \mathfrak{S} \\ (\text{croi}(\sigma), \text{decr}(\sigma)) \simeq J}} r_A(\mathbf{G}_\sigma) \quad (5.3.13)$$

$$= \sum_{\substack{\sigma \in \mathfrak{S} \\ (\text{croi}(\sigma), \text{decr}(\sigma)) \simeq J}} \sum_{\substack{u \in A^* \\ \text{std}(u) = \sigma}} u. \quad (5.3.14)$$

L'égalité entre (5.3.12) et (5.3.13) provient du lemme 4.3.7. L'égalité entre (5.3.14) et le membre droit de (5.3.10) provient du fait que $\text{croi}(\sigma) \simeq \text{croi}(u)$ (resp. $\text{decr}(\sigma) \simeq \text{decr}(u)$) lorsque $\text{std}(u) = \sigma$. \square

Une base booléenne

Law et Reading ont démontré dans [LR10] que la base par l'intermédiaire de laquelle ils ont défini leur algèbre de Hopf basée sur les permutations de Baxter tordues, analogue à la base fondamentale de **Baxter**, est une base booléenne. Nous donnons ici une autre démonstration de cette propriété en nous appuyant sur le critère établi dans le paragraphe 5.2.3.

Proposition 5.3.3. *La base fondamentale de **Baxter** est une base booléenne.*

Démonstration. Montrons que la relation d'équivalence sylvestre vérifie les hypothèses de la proposition 5.2.6. Le résultat suit immédiatement par le fait que d'après la proposition 4.2.8, la relation d'équivalence de Baxter est un raffinement de la relation sylvestre.

Commençons par établir un premier résultat intermédiaire. Soient x et y deux mots de même longueur sans répétition de lettre et $u, v \in x \sqcup y$, où les lettres de y sont décalées de $\max(x)$. Montrons par récurrence sur $|x| + |y|$ que si $\text{decr}(u)$ et $\text{decr}(v)$ sont des arbres binaires de même forme, alors $u = v$. Ceci est clair si $|x| + |y| = 0$. Sinon, nous avons $u = u' \cdot \mathbf{b} \cdot u''$ et $v = v' \cdot \mathbf{b} \cdot v''$ où $\mathbf{b} := \max(u) = \max(v)$. Comme la forme du sous-arbre gauche de $\text{decr}(u)$ est égale à la forme du sous-arbre gauche de $\text{decr}(v)$, la position de \mathbf{b} est la même dans u et dans v . De plus, y est de la forme $y = y' \cdot \mathbf{a} \cdot y''$ où $\mathbf{a} := \max(y)$, et x est de la forme $x = x' \cdot x''$ où $u', v' \in x' \sqcup y'$ et $u'', v'' \in x'' \sqcup y''$. Comme le sous-arbre gauche (resp. droit) de $\text{decr}(u)$ est égal au sous-arbre gauche (resp. droit) de $\text{decr}(v)$, nous avons, par hypothèse de récurrence $u' = v'$ et $u'' = v''$, montrant $u = v$.

Soient maintenant $\pi, \mu \in \mathfrak{S}$ et $\sigma \neq \nu \in \pi \sqcup \mu$, et supposons $\sigma^{-1} \equiv_s \nu^{-1}$. Alors, d'après le théorème 4.1.6, les permutations σ^{-1} et ν^{-1} donnent le même arbre binaire de recherche lorsqu'elles sont insérées de la droite vers la gauche. D'après le lemme 4.3.7, ceci implique que $\text{decr}(\sigma)$ et $\text{decr}(\nu)$ sont de même forme. Nous avons par conséquent $\sigma = \nu$, ce qui contredit notre hypothèse. \square

L'algèbre de Hopf duale

L'algèbre de Hopf \mathbf{Baxter}^* , duale de \mathbf{Baxter} , est obtenue selon la description générale des duales des sous-algèbres de Hopf de \mathbf{FQSym} obtenue à partir de bons monoïdes donnée dans le paragraphe 5.1.1.

Notons $B(J)$ le mot calculé par l'algorithme EXTRAIREBAXTER (dont la définition figure dans le paragraphe 4.3.3 du chapitre 4) sur l'entrée du couple d'arbres binaires de recherche jumeaux J . Rappelons que lorsque J est étiqueté par une permutation, $B(J)$ est une permutation de Baxter.

D'après la définition générale du produit de \mathbf{MFQSym}^* donnée en (5.1.16), nous obtenons l'expression suivante pour le produit de \mathbf{Baxter}^* :

$$\mathbf{P}_{J_0}^* \cdot \mathbf{P}_{J_1}^* = \sum_{\substack{u \cdot v \in \mathfrak{S} \\ \text{std}(u) = B(J_0) \\ \text{std}(v) = B(J_1)}} \mathbf{P}_{\mathbb{P}(u \cdot v)}^* \quad (5.3.15)$$

Par exemple,

$$\begin{aligned} \mathbf{P}^* \cdot \mathbf{P}^* &= \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* \\ &+ \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* + \mathbf{P}^* \end{aligned} \quad (5.3.16)$$

De même, d'après la définition générale du coproduit de \mathbf{MFQSym}^* donnée en (5.1.18), nous obtenons l'expression suivante pour le coproduit de \mathbf{Baxter}^* :

$$\Delta(\mathbf{P}_J^*) = \sum_{\substack{0 \leq i \leq |J| \\ \sigma := B(J)_{|[1, i]} \\ \nu := \text{std}(B(J)_{|[i+1, |\pi|]})}} \mathbf{P}_{\mathbb{P}(\sigma)}^* \otimes \mathbf{P}_{\mathbb{P}(\nu)}^* \quad (5.3.17)$$

Par exemple,

$$\begin{aligned} \Delta\left(\mathbf{P}^*\right) &= 1 \otimes \mathbf{P}^* + \mathbf{P}^* \otimes \mathbf{P}^* + \mathbf{P}^* \otimes \mathbf{P}^* \\ &+ \mathbf{P}^* \otimes \mathbf{P}^* + \mathbf{P}^* \otimes 1. \end{aligned} \quad (5.3.18)$$

Remarquons que la base des \mathbf{P}^* de \mathbf{Baxter}^* est, par dualité, une base booléenne car d'après la proposition 5.3.3, la base des \mathbf{P} de \mathbf{Baxter} l'est. Ainsi, et d'après (5.3.15), le nombre d'éléments qui apparaissent dans un produit $\mathbf{P}_{J_0}^* \cdot \mathbf{P}_{J_1}^*$ dépend uniquement des tailles n et m de J_0 et J_1 et est $\binom{n+m}{n}$. De manière similaire, et d'après (5.3.17), le nombre d'éléments qui apparaissent dans un coproduit $\Delta(\mathbf{P}_J^*)$ dépend uniquement de la taille n de J et est $n + 1$.

Suivant la théorie de Fomin [Fom94], il est possible de construire une *paire de graphes gradués en dualité* $(G_{\mathbf{P}}, G_{\mathbf{P}^*})$ (voir aussi [HN07] et [Nze08]). L'ensemble des sommets de $G_{\mathbf{P}}$ et $G_{\mathbf{P}^*}$ est l'ensemble des couples d'arbres binaires de recherche jumeaux étiquetés par une permutation. Il y a un arc entre les sommets J et J' dans $G_{\mathbf{P}}$ (resp. dans $G_{\mathbf{P}^*}$) si $\mathbf{P}_{J'}$ (resp. $\mathbf{P}_{J'}^*$) apparaît dans le produit $\mathbf{P}_J \cdot \mathbf{P}_{\bullet\bullet}$ (resp. $\mathbf{P}_J^* \cdot \mathbf{P}_{\bullet\bullet}^*$). La figure 5.1 (resp. la figure 5.2) montre le graphe gradué $G_{\mathbf{P}}$ (resp. $G_{\mathbf{P}^*}$) restreint aux sommets de taille inférieure à quatre.

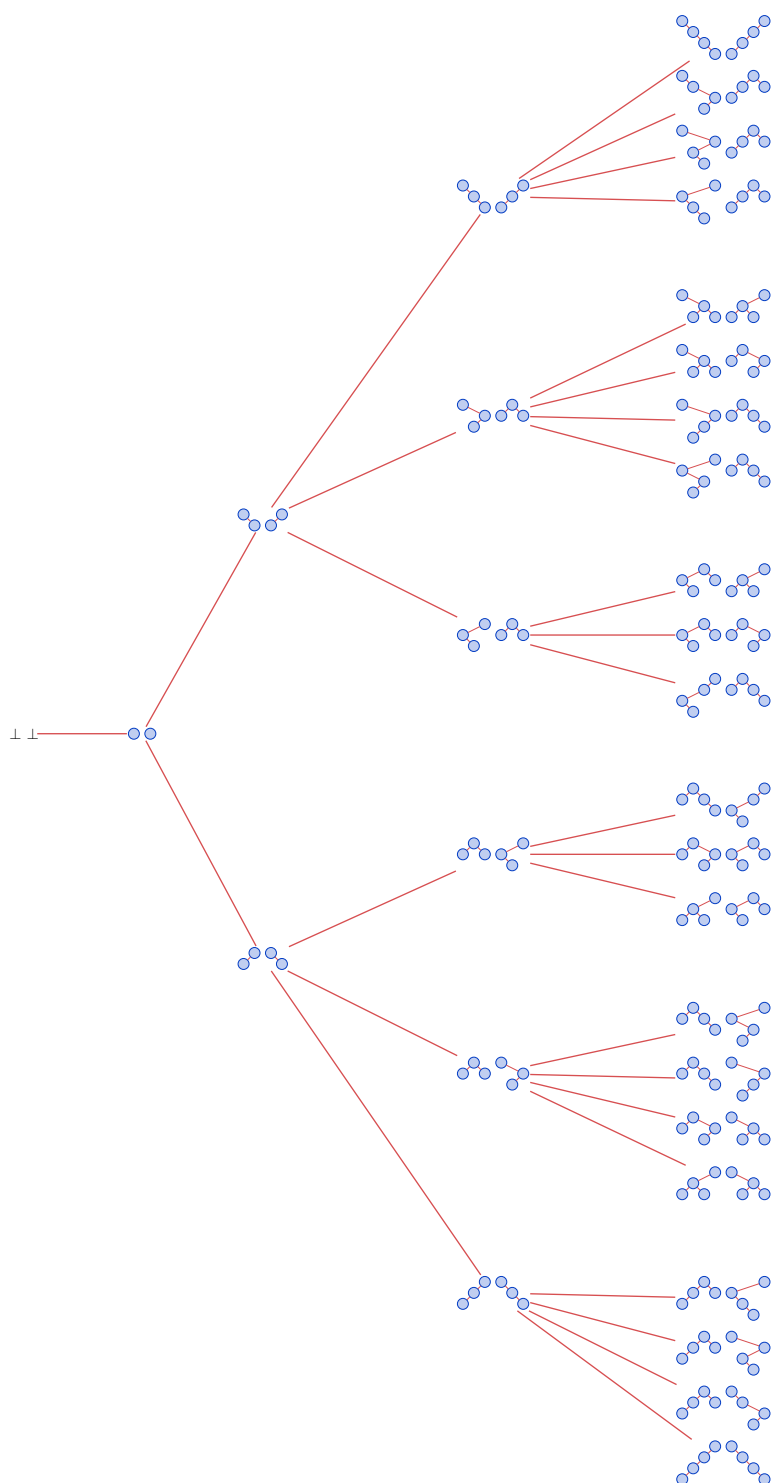


FIGURE 5.1 – Le graphe gradué G_P associé à **Baxter** et restreint aux sommets de taille inférieure à quatre.

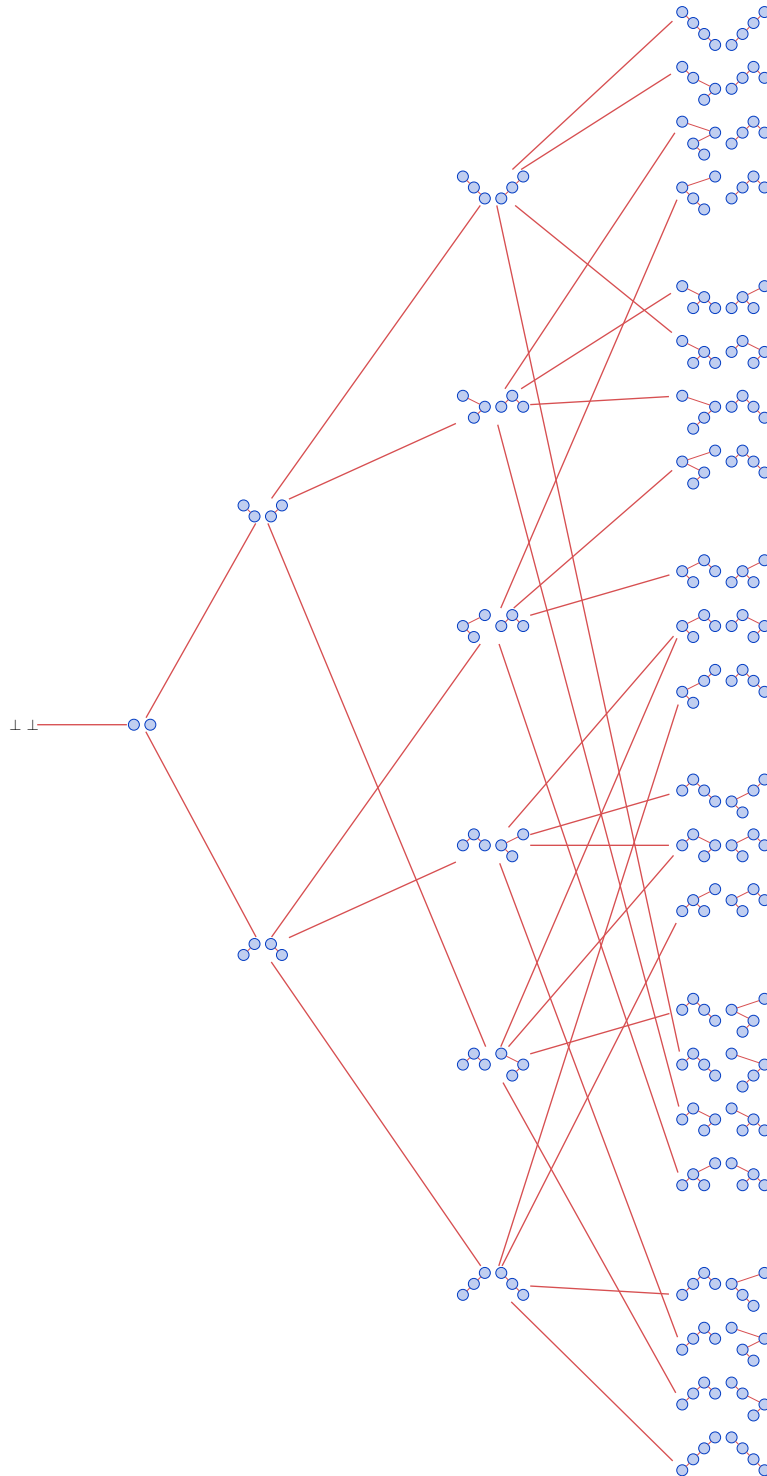


FIGURE 5.2 – Le graphe gradué $G_{\mathbf{P}^*}$ associé à \mathbf{Baxter}^* et restreint aux sommets de taille inférieure à quatre.

Produit et treillis de Baxter

Nous introduisons les opérateurs $/$ et \setminus sur l'ensemble des couples d'arbres binaires jumeaux. Ils sont définis, pour tous couples d'arbres binaires jumeaux $J_0 = (T_L^0, T_R^0)$ et $J_1 = (T_L^1, T_R^1)$ par :

$$J_0 / J_1 := (T_L^0 \setminus T_L^1, T_R^0 / T_R^1), \quad (5.3.19)$$

et

$$J_0 \setminus J_1 := (T_L^0 / T_L^1, T_R^0 \setminus T_R^1), \quad (5.3.20)$$

où les opérateurs $/$ et \setminus sur les arbres binaires sont respectivement les opérateurs *over* et *under* de Loday et Ronco [LR02]. Il est clair que J_0 / J_1 est un couple d'arbres binaires jumeaux puisque la canopée de $T_L^0 \setminus T_L^1$ est de la forme $\text{cnp}(T_L^0) \cdot 1 \cdot \text{cnp}(T_L^1)$ et celle de T_R^0 / T_R^1 est de la forme $\text{cnp}(T_R^0) \cdot 0 \cdot \text{cnp}(T_R^1)$. Il en est de même pour $J_0 \setminus J_1$.

Voici un lemme qui nous sera utile dans ce qui suit :

Lemme 5.3.4. *Pour toutes permutations σ et ν , nous avons*

$$\mathbb{P}(\sigma / \nu) = \mathbb{P}(\sigma) / \mathbb{P}(\nu), \quad (5.3.21)$$

et

$$\mathbb{P}(\sigma \setminus \nu) = \mathbb{P}(\sigma) \setminus \mathbb{P}(\nu). \quad (5.3.22)$$

Démonstration. C'est une conséquence directe de la définition 4.3.1 du \mathbb{P} -symbole de Baxter, des définitions des opérateurs $/$ et \setminus sur les permutations (dont la définition est donnée en (1.2.13) et (1.2.14) dans le chapitre 1) et des définitions en (5.3.19) et (5.3.20) des opérateurs $/$ et \setminus sur les couples d'arbres binaires jumeaux. \square

Le lemme 5.3.4 implique en particulier que toute classe d'équivalence de Baxter de la forme $\hat{\sigma} / \hat{\nu}$ (resp. $\hat{\sigma} \setminus \hat{\nu}$) peut être encodée par le couple d'arbres binaires jumeaux $\mathbb{P}(\sigma) / \mathbb{P}(\nu)$ (resp. $\mathbb{P}(\sigma) \setminus \mathbb{P}(\nu)$) où $\sigma \in \hat{\sigma}$ et $\nu \in \hat{\nu}$.

Les opérateurs $/$ et \setminus sur l'ensemble des couples d'arbres binaires jumeaux jouent un rôle analogue aux opérateurs de Loday et Ronco sur les arbres binaires. En effet,

Proposition 5.3.5. *Le produit dans la base fondamentale de **Baxter** vérifie, pour tous couples d'arbres binaires jumeaux J_0 et J_1 ,*

$$\mathbf{P}_{J_0} \cdot \mathbf{P}_{J_1} = \sum_{J_0 / J_1 \leq_B J \leq_B J_0 \setminus J_1} \mathbf{P}_J. \quad (5.3.23)$$

Démonstration. Étant donné que, d'après le théorème 4.4.3, la relation d'équivalence de Baxter est une congruence du permutoèdre, nous pouvons appliquer la proposition 5.2.2 sur **Baxter**, montrant que les éléments de sa base fondamentale qui apparaissent dans un produit forment un intervalle du treillis de Baxter. Le résultat suit du lemme 5.3.4. \square

Bases multiplicatives et générateurs libres

En nous appuyant sur le fait que la relation d'équivalence de Baxter est une congruence du permutoèdre (voir le théorème 4.4.3) et suivant le paragraphe 5.2.2, nous définissons les bases suivantes de **Baxter**. La *base élémentaire* est définie par

$$\mathbf{E}_J := \sum_{J \leq_B J'} \mathbf{P}_{J'}, \quad (5.3.24)$$

et la *base homogène* est définie par

$$\mathbf{H}_J := \sum_{J' \leq_B J} \mathbf{P}_{J'}. \quad (5.3.25)$$

Proposition 5.3.6. *Le produit dans la base élémentaire de **Baxter** vérifie, pour tous couples d'arbres binaires jumeaux J_0 et J_1 ,*

$$\mathbf{E}_{J_0} \cdot \mathbf{E}_{J_1} = \mathbf{E}_{J_0 / J_1}. \quad (5.3.26)$$

*De même, le produit dans la base homogène de **Baxter** vérifie*

$$\mathbf{H}_{J_0} \cdot \mathbf{H}_{J_1} = \mathbf{H}_{J_0 \setminus J_1}. \quad (5.3.27)$$

Démonstration. Étant donné que, d'après le théorème 4.4.3, la relation d'équivalence de Baxter est une congruence du permutoèdre, nous pouvons appliquer la proposition 5.2.4 sur **Baxter**. Le résultat suit du lemme 5.3.4. \square

Voici quelques exemples de produits :

$$\mathbf{E} \cdot \mathbf{E} = \mathbf{E}, \quad (5.3.28)$$

$$\mathbf{H} \cdot \mathbf{H} = \mathbf{H}. \quad (5.3.29)$$

La proposition 5.3.6 montre également que les bases des \mathbf{E} et des \mathbf{H} de **Baxter** sont des bases booléennes d'algèbre. Elles ne sont cependant pas des bases booléennes de cogèbre puisque l'on a

$$\Delta(\mathbf{E}_{\text{arbre}}) = 1 \otimes \mathbf{E}_{\text{arbre}} + 2 \mathbf{E}_{\text{arbre}} \otimes \mathbf{E}_{\text{arbre}} + \mathbf{E}_{\text{arbre}} \otimes 1, \quad (5.3.30)$$

et

$$\Delta(\mathbf{H}_{\text{arbre}}) = 1 \otimes \mathbf{H}_{\text{arbre}} + 2 \mathbf{H}_{\text{arbre}} \otimes \mathbf{H}_{\text{arbre}} + \mathbf{H}_{\text{arbre}} \otimes 1. \quad (5.3.31)$$

Un couple d'arbres binaires jumeaux J est *connexe* (resp. *anti-connexe*) si toutes les permutations σ telles que $\mathbb{P}(\sigma) = J$ sont connexes (resp. anti-connexes).

Proposition 5.3.7. *L'algèbre **Baxter** est engendrée librement par les éléments \mathbf{E}_J (resp. \mathbf{H}_J) tels que J est un couple d'arbres binaires jumeaux connexe (resp. anti-connexe).*

Démonstration. Étant donné que d'après le théorème 4.4.3, la relation d'équivalence de Baxter est une congruence du permutoèdre, nous pouvons appliquer la proposition 5.2.5 sur **Baxter**. Le résultat suit du fait que la notion de connexité (resp. d'anti-connexité) sur les couples d'arbres binaires jumeaux coïncide par définition avec la notion de connexité des classes d'équivalence de Baxter. \square

Soit $B_C(t)$ la série génératrice des couples d'arbres binaires jumeaux connexes (resp. anti-connexes). La proposition 5.3.7 implique que la série de Hilbert $B(t)$ de **Baxter** vérifie $B(t) = 1/(1 - B_C(t))$. Par conséquent, $B_C(z)$ vérifie

$$B_C(t) = 1 - \frac{1}{B(t)}. \quad (5.3.32)$$

Les premières dimensions des générateurs algébriques de **Baxter** sont

$$0, 1, 1, 3, 11, 47, 221, 1113, 5903, 32607, 186143, 1092015. \quad (5.3.33)$$

Voici les générateurs algébriques de **Baxter**, d'ordre 1 à 4, et exprimés dans la base des **E** :

$$\mathbf{E}_{\bullet\bullet}; \quad (5.3.34)$$

$$\mathbf{E}_{\bullet\bullet\bullet}; \quad (5.3.35)$$

$$\mathbf{E}_{\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet}; \quad (5.3.36)$$

$$\begin{aligned} & \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \\ & \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}, \mathbf{E}_{\bullet\bullet\bullet\bullet\bullet}. \end{aligned} \quad (5.3.37)$$

Proposition 5.3.8. *Soit σ une permutation de Baxter connexe (resp. anti-connexe). Alors, toute permutation ν telle que $\nu \equiv_{\mathbf{B}} \sigma$ est également connexe (resp. anti-connexe).*

Démonstration. Comme toute permutation, σ s'écrit de manière unique sous la forme $\sigma = \sigma^{(1)} / \dots / \sigma^{(\ell)}$ où les permutations $\sigma^{(1)}, \dots, \sigma^{(\ell)}$ sont connexes. De plus, comme σ évite les motifs $2-41-3$ et $3-14-2$, toutes les permutations $\sigma^{(i)}$ les évitent aussi, ce qui implique que les $\sigma^{(i)}$ sont des permutations de Baxter. Cette remarque montre que la série génératrice des permutations de Baxter connexes est $B_C(t)$, et par conséquent, que les permutations de Baxter connexes de taille n et les couples d'arbres binaires jumeaux connexes de taille n sont en bijection. Le résultat suit du théorème 4.3.13 disant que toute classe d'équivalence de Baxter de permutations contient exactement une permutation de Baxter. La démonstration dans le cas anti-connexe est analogue. \square

La proposition 5.3.8 fournit ainsi une autre description des générateurs libres de **Baxter** :

Corollaire 5.3.9. *L'algèbre **Baxter** est engendrée librement par les éléments \mathbf{E}_J (resp. \mathbf{H}_J) tels que la permutation de Baxter appartenant à la classe d'équivalence de Baxter encodée par J est connexe (resp. anti-connexe).*

Structure bidendriforme et autodualité

Il est évident d'après la définition des relations d'adjacence de Baxter (voir (4.2.2) et (4.2.3) dans le chapitre 4) que tous les mots d'une classe d'équivalence de Baxter se terminent par une même lettre. Ainsi, d'après ce que nous avons exposé dans le paragraphe 5.2.4, **Baxter** est une sous-bigèbre bidendriforme de **FQSym**. Ceci implique donc le résultat suivant :

Proposition 5.3.10. *L'algèbre de Hopf **Baxter** est autoduale, libre, colibre, l'algèbre de Lie engendrée par ses éléments primitifs est libre et l'algèbre dendriforme engendrée par ses éléments totalement primitifs l'est également.*

Démonstration. Ce résultat suit de la proposition 5.2.11. \square

Étant donné que d'après la proposition 5.3.10, **Baxter** est autoduale, il est naturel de se demander si le morphisme de Hopf analogue à l'isomorphisme $\tau \circ \psi \circ \theta$ entre **PBT** et **PBT**^{*} rappelé dans le paragraphe 5.1.3 est un isomorphisme. En notant $\theta : \mathbf{Baxter} \hookrightarrow \mathbf{FQSym}$

l'injection de **Baxter** dans **FQSym** définie suivant (5.3.1), le morphisme de Hopf $\tau \circ \psi \circ \theta : \mathbf{Baxter} \rightarrow \mathbf{Baxter}^*$ n'est pas un isomorphisme. En effet,

$$\tau \circ \psi \circ \theta \left(\mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} \right) = \tau \circ \psi (\mathbf{F}_{2143} + \mathbf{F}_{2413}) = \tau (\mathbf{F}_{2143}^* + \mathbf{F}_{3142}^*) = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}^* + \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}^*, \quad (5.3.38)$$

$$\tau \circ \psi \circ \theta \left(\mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} \right) = \tau \circ \psi (\mathbf{F}_{3142} + \mathbf{F}_{3412}) = \tau (\mathbf{F}_{2413}^* + \mathbf{F}_{3412}^*) = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}^* + \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}^*, \quad (5.3.39)$$

montrant que $\tau \circ \psi \circ \theta$ n'est pas injectif.

Éléments primitifs et totalement primitifs

Étant donné que la famille des éléments \mathbf{E}_J (resp. \mathbf{H}_J) où J est un couple d'arbres binaires jumeaux connexe (resp. anti-connexe) sont les éléments indécomposables de **Baxter**, la famille duale \mathbf{E}_J^* (resp. \mathbf{H}_J^*) forme une base de l'algèbre de Lie des éléments primitifs de **Baxter**^{*}. De plus, d'après la proposition 5.3.10, cette algèbre de Lie est libre.

D'après les résultats de Foissy [Foi07], la série génératrice $B_T(t)$ des éléments totalement primitifs de **Baxter** est

$$B_T(t) = \frac{B(t) - 1}{B(t)^2}. \quad (5.3.40)$$

Les premières dimensions des éléments totalement primitifs de **Baxter** sont

$$0, 1, 0, 1, 4, 19, 96, 511, 2832, 16215, 95374, 573837. \quad (5.3.41)$$

Voici des bases d'ordre 1, 3 et 4 des éléments totalement primitifs de **Baxter** :

$$t_{1,1} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}, \quad (5.3.42)$$

$$t_{3,1} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}, \quad (5.3.43)$$

$$t_{4,1} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} + \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} + \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} + \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}, \quad (5.3.44)$$

$$t_{4,2} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}, \quad (5.3.45)$$

$$t_{4,3} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}, \quad (5.3.46)$$

$$t_{4,4} = \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}} - \mathbf{P}_{\begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array}}. \quad (5.3.47)$$

Compatibilité avec le produit

Aval et Viennot ont introduit [AV10] un nouveau produit associatif dans **PBT**, appelé *produit #*. Le produit de deux éléments de degrés respectifs n et m est un élément de degré $n+m-1$. Aval, Novelli et Thibon ont généralisé ce produit au niveau de l'algèbre associative et montré qu'il reste bien défini dans **FQSym** [ANT11].

Le produit $\#$ s'exprime dans \mathbf{FQSym} de la manière suivante. Soient pour tout $k \geq 1$ les applications linéaires $d_k : \mathbf{FQSym}^{(n)} \rightarrow \mathbf{FQSym}^{(n-1)}$ définies pour toute permutation σ de taille n par

$$d_k(\mathbf{F}_\sigma) := \begin{cases} \mathbf{F}_{\text{std}(\sigma_1 \dots \sigma_i \sigma_{i+2} \dots \sigma_n)} & \text{s'il existe } i \in [n-1] \text{ tel que } \sigma_i = k \text{ et } \sigma_{i+1} = k+1, \\ 0 & \text{sinon.} \end{cases} \quad (5.3.48)$$

On définit maintenant linéairement pour toutes permutations σ et ν ,

$$\mathbf{F}_\sigma \# \mathbf{F}_\nu := d_n(\mathbf{F}_\sigma \cdot \mathbf{F}_\nu), \quad (5.3.49)$$

où n est la taille de σ .

Proposition 5.3.11. *Les applications linéaires d_k sont bien définies dans **Baxter**. Plus précisément, nous avons pour tout couple d'arbres binaires jumeaux $J := (T_0, T_1)$,*

$$d_k(\mathbf{P}_J) = \begin{cases} \mathbf{P}_{J'} & \text{si le } (k+1)^e \text{ (resp. } k^e \text{) nœud est fils du } k^e \text{ (resp. } (k+1)^e \text{) nœud dans } T_L \text{ (resp. } T_R), \\ 0 & \text{sinon,} \end{cases} \quad (5.3.50)$$

où $J' := (T'_L, T'_R)$ est le couple d'arbres binaires jumeaux obtenu en contractant dans T_L et T_R les arcs connectant les k^e et $(k+1)^e$ nœuds.

Démonstration. Nous nous basons sur le fait que, en vertu du paragraphe 4.3.3 du chapitre 4, les permutations d'une classe de Baxter encodée par (T_L, T_R) sont exactement les extensions linéaires des posets $\Delta(T_L)$ et $\nabla(T_R)$.

Nous avons deux cas à explorer suivant si le $(k+1)^e$ (resp. k^e) nœud est fils du k^e (resp. $(k+1)^e$) nœud dans T_L (resp. T_R).

Cas 1. Si c'est le cas, il existe dans la classe encodée par J des permutations qui possèdent un facteur $k \cdot k + 1$. L'application d_k efface les lettres de valeur $k+1$ dans ces permutations et les standardise. Les permutations ainsi obtenues sont exactement les extensions linéaires des posets $\Delta(T'_L)$ et $\nabla(T'_R)$.

Cas 2. Lorsque ce n'est pas le cas, comme les k^e nœuds et $(k+1)^e$ nœuds d'un arbre binaire sont en relation de parenté, aucune permutation de la classe de Baxter encodée par J ne possède de facteur $k \cdot k + 1$. Ceci implique $d_k(\mathbf{P}_J) = 0$. \square

Nous avons par exemple

$$d_3 \left(\mathbf{P}_{\text{arbre}} \right) = \mathbf{P}_{\text{arbre}}. \quad (5.3.51)$$

La proposition 5.3.11 montre en particulier que le produit $\#$ est interne à **Baxter**.

Liens avec d'autres sous-algèbres de Hopf de \mathbf{FQSym}

D'après la proposition 4.2.8, toute classe d'équivalence sylvestre est une union de classes d'équivalence de Baxter. Ce résultat implique l'existence du morphisme de Hopf injectif

$$\rho : \mathbf{PBT} \hookrightarrow \mathbf{Baxter}, \quad (5.3.52)$$

vérifiant

$$\rho(\mathbf{P}_T) = \sum_{\substack{T' \in \mathcal{T} \\ J := (T', T) \in \mathcal{J}}} \mathbf{P}_J, \quad (5.3.53)$$

pour tout arbre binaire T . Par exemple,

$$\rho \left(\mathbf{P} \begin{array}{c} \bullet \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} \right) = \mathbf{P} \begin{array}{c} \bullet \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \mathbf{P} \begin{array}{c} \bullet \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \mathbf{P} \begin{array}{c} \bullet \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array}. \quad (5.3.54)$$

La relation d'équivalence \equiv_{3R} mène à la construction de la sous-algèbre de Hopf $\mathbf{DSym}^{(3)}$ (voir [NRT09]) de \mathbf{FQSym} , où sa base fondamentale est définie selon (5.1.1). D'après la proposition 4.2.11, toute classe d'équivalence de \equiv_{3R} de permutations est une union de classes d'équivalence de Baxter. Ce résultat implique l'existence du morphisme de Hopf injectif

$$\alpha : \mathbf{DSym}^{(3)} \hookrightarrow \mathbf{Baxter}, \quad (5.3.55)$$

vérifiant

$$\alpha (\mathbf{P}_{\hat{\sigma}}) = \sum_{\sigma \in \hat{\sigma} \cap \mathfrak{S}^B} \mathbf{P}_{\mathbb{P}(\sigma)}, \quad (5.3.56)$$

pour toute classe d'équivalence $\hat{\sigma}$ de permutations de \equiv_{3R} .

Nous pouvons composer les morphismes de Hopf injectifs $\beta : \mathbf{Sym} \hookrightarrow \mathbf{PBT}$ (défini dans le paragraphe 5.1.4) et $\rho : \mathbf{PBT} \hookrightarrow \mathbf{Baxter}$ pour obtenir le morphisme de Hopf injectif

$$\gamma : \mathbf{Sym} \hookrightarrow \mathbf{Baxter}. \quad (5.3.57)$$

Celui-ci vérifie

$$\gamma (\mathbf{P}_b) = \sum_{\substack{(T_L, T_R) \in \mathcal{J} \\ \text{cnp}(T_R) = b}} \mathbf{P}_J, \quad (5.3.58)$$

pour tout mot b sur l'alphabet $\{0, 1\}$.

Les liens entre les sous-algèbres de Hopf connues de \mathbf{FQSym} et l'algèbre de Hopf \mathbf{Baxter} sont résumés par le diagramme de la figure 5.3.

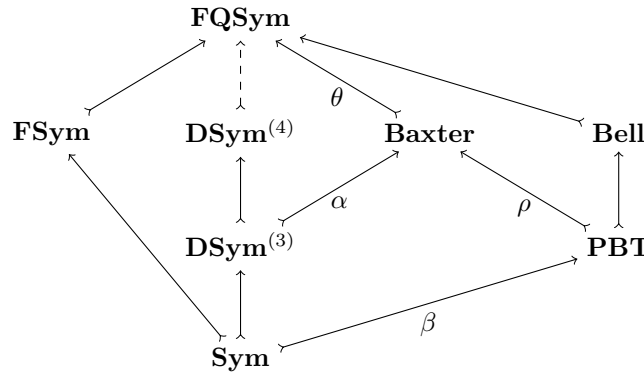


FIGURE 5.3 – Le diagramme des sous-algèbres de Hopf connues de \mathbf{FQSym} . Les flèches \hookrightarrow sont des injections d'algèbres de Hopf.

Troisième partie

Monoides, opérades et algèbres de Hopf combinatoires

Chapitre 6

Construction d'opérades à partir de monoïdes

Plusieurs opérades qui mettent en jeu des objets combinatoires furent introduites et étudiées ces dernières années. Additionnellement aux exemples classiques que constituent l'opérade associative, l'opérade commutative et associative et l'opérade de Lie, la plupart des opérades d'introduction récente portent sur divers objets combinatoires. Beaucoup portent en particulier sur diverses espèces d'arbres [Lod01], [CL01], [BF03], [Cha04], [Cha06b], [Liv06], [Cha10]. Ces opérades sont construites de manière plus ou moins indépendante les unes par rapport aux autres et ils ne semble pas qu'il existe de construction unificatrice pour les obtenir toutes et les étudier d'un même tenant.

L'objectif de ce chapitre est d'introduire un procédé de construction d'opérades qui se veut, d'un certain point de vue, assez général. Le point de départ de cette construction consiste en un monoïde M quelconque. On considère alors l'ensemble des mots dont les lettres sont des éléments de M . L'arité d'un élément est sa longueur, et cet ensemble est muni d'une substitution partielle dont la définition est dictée par l'expression du produit de M . Cet ensemble est également muni d'une action du groupe symétrique où ce dernier agit sur un mot en permutant ses lettres. La structure ainsi obtenue est une opérade ensembliste.

L'aspect combinatoire de cette construction s'illustre par le fait que ce procédé donne gratuitement de nombreuses nouvelles opérades dont certaines séries de Hilbert sont connues en combinatoire. Ainsi, en choisissant une bijection appropriée entre les éléments de l'opérade et une classe combinatoire qui possède la même série génératrice, la substitution partielle et l'action du groupe symétrique peuvent se réinterpréter en termes d'opérations sur ces objets combinatoires.

Ce travail encore en cours s'appuie sur une grande part d'expérimentation informatique. Nous avons en effet à ce titre développé plusieurs programmes sur Sage [S⁺11] qui implantent notre construction. Ainsi, sur l'entrée d'un monoïde, l'opérade associée est instanciée et nous pouvons lui poser diverses questions, comme par exemple les premiers coefficients de la série de Hilbert d'une sous-opérade engendrée par une famille finie G de générateurs, afficher les éléments engendrés par G arité par arité ou encore calculer les relations non triviales entre les éléments de G .

Comme cela est mentionné dans le paragraphe 3.3 du chapitre 3, une opérade permet de construire un groupe et deux algèbres de Hopf, l'une commutative et l'autre non. Ainsi, en composant cette construction avec la nôtre, il est possible d'obtenir un groupe et deux algèbres de Hopf à partir d'un monoïde. Nous considérons ici l'algèbre de Hopf non commutative obtenue

à partir de l'une des opérades que nous construisons. Celle-ci est basée sur des forêts ordonnées d'arbres plans enracinés où le degré d'un élément est la donnée de son nombre d'arêtes. Cette algèbre de Hopf est très similaire à l'algèbre de Hopf de Connes-Kreimer (voir le paragraphe 2.2.2 du chapitre 2) notamment dans l'expression de son coproduit.

Ce chapitre est organisé de la manière suivante. Le paragraphe 6.1 est consacré à la définition de l'ingrédient principal du chapitre, à savoir la construction qui à un monoïde associe une opérade. Nous montrons que cette construction est un foncteur de la catégorie des monoïdes vers la catégorie des opérades et que celui-ci respecte les injections et les surjections. Nous développons ensuite dans le paragraphe 6.2 plusieurs exemples de sous-opérades, symétriques ou non, de l'opérade obtenue à partir du monoïde additif. Les opérades que nous présentons mettent en jeu des objets combinatoires relativement variés : endofonctions, fonctions de parking, mots tassés, permutations, arbres plans enracinés, k -chemins de Dyck, arbres de Schröder, chemins de Motzkin, compositions d'entiers, animaux dirigés et compositions d'entiers segmentées. Ces opérades s'organisent dans un diagramme dont les flèches sont des injections ou des surjections et certaines sont de plus finement engendrées. Nous retrouvons aussi l'opérade diassociative [Lod01], [Zin10] comme sous-opérade de l'opérade obtenue à partir du monoïde multiplicatif. Nous terminons par le paragraphe 6.3 en étudiant l'algèbre de Hopf non commutative obtenue à partir de l'opérade sur les arbres plans enracinés définie dans le paragraphe précédent. Nous proposons une réalisation polynomiale de cette algèbre de Hopf en utilisant des procédés inspirés de [FNT10] introduits pour la réalisation polynomiale de l'algèbre de Hopf de Connes-Kreimer.

6.1 Un foncteur des monoïdes vers les opérades ensemblistes

Nous décrivons ici une construction qui à un monoïde M associe une opérade ensembliste dont les éléments sont des mots sur M vu comme un alphabet. Nous montrons ensuite quelques propriétés de cette construction et en particulier qu'elle est fonctorielle.

6.1.1 La construction

Définition 6.1.1. La transformation T associe à tout monoïde $(M, \bullet, 1)$ la structure $\mathsf{TM} := (\mathsf{TM}, \circ, \mathbf{1}, \cdot)$ définie de la manière suivante. L'ensemble TM est de la forme

$$\mathsf{TM} := \bigsqcup_{n \geq 1} \mathsf{TM}(n), \quad (6.1.1)$$

où, pour tout $n \geq 1$,

$$\mathsf{TM}(n) := \{(x_1, \dots, x_n) : x_i \in M \text{ pour tout } i \in [n]\}. \quad (6.1.2)$$

Le symbole \circ est un opérateur de substitution partielle

$$\circ_i : \mathsf{TM}(n) \times \mathsf{TM}(m) \rightarrow \mathsf{TM}(n + m - 1), \quad (6.1.3)$$

qui est défini pour tous $x \in \mathsf{TM}(n)$, $y \in \mathsf{TM}(m)$ et $i \in [n]$ par

$$x \circ_i y := (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_n). \quad (6.1.4)$$

L'élément $\mathbf{1} \in \mathsf{TM}(1)$ est défini par

$$\mathbf{1} := (1). \quad (6.1.5)$$

Le symbole \cdot est une action du groupe symétrique

$$\cdot : \mathsf{TM}(n) \times \mathfrak{S}_n \rightarrow \mathsf{TM}(n), \quad (6.1.6)$$

qui est définie pour tous $x \in \mathsf{TM}(n)$ et $\sigma \in \mathfrak{S}_n$ par

$$x \cdot \sigma := (x_{\sigma_1}, \dots, x_{\sigma_n}). \quad (6.1.7)$$

Soit M un monoïde. Les éléments de TM sont des mots dont les lettres sont des éléments de M . La *longueur* d'un élément x de $\mathsf{TM}(n)$, notée $|x|$, est l'entier n . Pour des questions de lisibilité, nous noterons dans certains cas un élément $(x_1, \dots, x_n) \in \mathsf{TM}(n)$ sous forme de mot $x_1 \dots x_n$.

Proposition 6.1.2. *Soit M un monoïde. Alors, la structure TM est une opérade ensembliste.*

Démonstration. Soient \bullet le produit et 1 l'unité du monoïde M .

Tout d'abord, le domaine de définition de l'opérateur de substitution partielle \circ est bien conforme à (6.1.3). En effet, si $x \in \mathsf{TM}(n)$, $y \in \mathsf{TM}(m)$ et $i \in [n]$, l'élément $z := x \circ_i y$ est tel que $|z| = |x| + |y| - 1$, et ainsi $z \in \mathsf{TM}(n + m - 1)$.

Montrons que la condition d'associativité en série (voir (3.1.5) dans le chapitre 3) est vérifiée. Soient $x \in \mathsf{TM}(n)$, $y \in \mathsf{TM}(m)$, $z \in \mathsf{TM}(k)$, $i \in [n]$ et $j \in [m]$. Nous avons, en se basant sur le fait que le produit \bullet de M est associatif,

$$\begin{aligned} (x \circ_i y) \circ_{i+j-1} z &= (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_n) \circ_{i+j-1} z \\ &= (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_{j-1}, (x_i \bullet y_j) \bullet z_1, \dots, (x_i \bullet y_j) \bullet z_k, \\ &\quad x_i \bullet y_{j+1}, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_n) \\ &= (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_{j-1}, x_i \bullet (y_j \bullet z_1), \dots, x_i \bullet (y_j \bullet z_k), \\ &\quad x_i \bullet y_{j+1}, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_n) \\ &= x \circ_i (y_1, \dots, y_{j-1}, y_j \bullet z_1, \dots, y_j \bullet z_k, y_{j+1}, \dots, y_m) \\ &= x \circ_i (y \circ_j z), \end{aligned} \quad (6.1.8)$$

ce qui montre cette relation.

Montrons maintenant que la condition d'associativité en parallèle (voir (3.1.6) dans le chapitre 3) est vérifiée. Soient $x \in \mathsf{TM}(n)$, $y \in \mathsf{TM}(m)$, $z \in \mathsf{TM}(k)$ et $i < j \in [n]$. Nous avons

$$\begin{aligned} (x \circ_j z) \circ_i y &= (x_1, \dots, x_{j-1}, x_j \bullet z_1, \dots, x_j \bullet z_k, x_{j+1}, \dots, x_n) \circ_i y, \\ &= (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_{j-1}, \\ &\quad x_j \bullet z_1, \dots, x_j \bullet z_k, x_{j+1}, \dots, x_n) \\ &= (x_1, \dots, x_{i-1}, x_i \bullet y_1, \dots, x_i \bullet y_m, x_{i+1}, \dots, x_n) \circ_{j+m-1} z \\ &= (x \circ_i y) \circ_{j+m-1} z, \end{aligned} \quad (6.1.9)$$

ce qui montre cette relation.

Montrons que l'élément $\mathbf{1}$ est bien l'unité de la substitution partielle \circ . Tout d'abord, nous avons bien $\mathbf{1} \in \mathsf{TM}(1)$. D'autre part, pour tous $x \in \mathsf{TM}(n)$ et $i \in [n]$, nous avons

$$x \circ_i \mathbf{1} = (x_1, \dots, x_{i-1}, x_i \bullet 1, x_{i+1}, \dots, x_n) = x, \quad (6.1.10)$$

car 1 est l'élément neutre du monoïde M . De plus,

$$\mathbf{1} \circ_1 x = (1 \bullet x_1, \dots, 1 \bullet x_n) = x, \quad (6.1.11)$$

pour la même raison, ce qui montre la propriété.

L'action du groupe symétrique \cdot sur TM respecte la relation d'équivariance (voir (3.1.9) dans le chapitre 3) car les éléments de \mathfrak{S}_n agissent par permutation des lettres des mots (x_1, \dots, x_n) de $\mathsf{TM}(n)$.

La structure TM est donc une opérade. Finalement, d'après la définition de sa substitution partielle, de son unité et de son action du groupe symétrique, cette opérade est de plus ensembliste. \square

D'après la proposition 6.1.2, la transformation T produit une opérade ensembliste sur la donnée d'un monoïde. Avec la définition suivante, T produit également une application sur l'entrée d'un morphisme de monoïdes.

Définition 6.1.3. Soient M et N deux monoïdes. La transformation T associe à tout morphisme de monoïdes $\theta : M \rightarrow N$ l'application

$$\mathsf{T}\theta : \mathsf{TM} \rightarrow \mathsf{TN} \quad (6.1.12)$$

définie, pour tout $(x_1, \dots, x_n) \in \mathsf{TM}(n)$, par

$$\mathsf{T}\theta(x_1, \dots, x_n) := (\theta(x_1), \dots, \theta(x_n)). \quad (6.1.13)$$

Proposition 6.1.4. Soient M et N deux monoïdes et $\theta : M \rightarrow N$ un morphisme de monoïdes. Alors, l'application $\mathsf{T}\theta : \mathsf{TM} \rightarrow \mathsf{TN}$ est un morphisme d'opérade.

Démonstration. Soient \bullet_M (resp. \bullet_N) et 1_M (resp. 1_N) le produit et l'unité du monoïde M (resp. N).

Soient $x \in \mathsf{TM}(n)$, $y \in \mathsf{TM}(m)$ et $i \in [n]$. En utilisant le fait que θ est un morphisme de monoïde, nous avons

$$\begin{aligned} \mathsf{T}\theta(x \circ_i y) &= \mathsf{T}\theta(x_1, \dots, x_{i-1}, x_i \bullet_M y_1, \dots, x_i \bullet_M y_m, x_{i+1}, \dots, x_n) \\ &= (\theta(x_1), \dots, \theta(x_{i-1}), \theta(x_i \bullet_M y_1), \dots, \theta(x_i \bullet_M y_m), \\ &\quad \theta(x_{i+1}), \dots, \theta(x_n)) \\ &= (\theta(x_1), \dots, \theta(x_{i-1}), \theta(x_i) \bullet_N \theta(y_1), \dots, \theta(x_i) \bullet_N \theta(y_m), \\ &\quad \theta(x_{i+1}), \dots, \theta(x_n)) \\ &= (\theta(x_1), \dots, \theta(x_n)) \circ_i (\theta(y_1), \dots, \theta(y_m)) \\ &= \mathsf{T}\theta(x) \circ_i \mathsf{T}\theta(y). \end{aligned} \quad (6.1.14)$$

Par définition de la transformation T , (1_M) est l'unité de TM , et nous avons

$$\mathsf{T}\theta(1_M) = (\theta(1_M)) = (1_N). \quad (6.1.15)$$

Finalement, pour tout $x \in \mathsf{TM}(n)$ et $\sigma \in \mathfrak{S}_n$, on a clairement $\mathsf{T}\theta(x \cdot \sigma) = \mathsf{T}\theta(x) \cdot \sigma$. L'application $\mathsf{T}\theta$ respecte ainsi les trois propriétés requises, et est ainsi un morphisme d'opérades. \square

6.1.2 Propriétés de la construction

Théorème 6.1.5. La transformation T est un foncteur de la catégorie des monoïdes avec morphismes de monoïde vers la catégorie des opérades ensemblistes avec morphismes d'opérade.

Démonstration. Remarquons tout d'abord que par la proposition 6.1.2, la transformation T construit bien une opérade ensembliste à partir d'un monoïde et par la proposition 6.1.4, un morphisme d'opérade à partir d'un morphisme de monoïde.

Soient M un monoïde, $\theta : M \rightarrow M$ le morphisme identité associé à M et x un élément de $TM(n)$. Nous avons

$$T\theta(x) = (\theta(x_1), \dots, \theta(x_n)) = (x_1, \dots, x_n) = x, \quad (6.1.16)$$

ce qui montre que $T\theta$ est bien le morphisme identité d'opérade associé à l'opérade TM .

Soient (L, \bullet_L) , (M, \bullet_M) et (N, \bullet_N) trois monoïdes, $\theta : L \rightarrow M$ et $\omega : M \rightarrow N$ deux morphismes de monoïde et x un élément de $TL(n)$. Nous avons alors

$$\begin{aligned} T(\omega \circ \theta)(x) &= (\omega \circ \theta(x_1), \dots, \omega \circ \theta(x_n)) \\ &= (\omega(\theta(x_1)), \dots, \omega(\theta(x_n))) \\ &= T\omega(\theta(x_1), \dots, \theta(x_n)) \\ &= (T\omega \circ T\theta)(x), \end{aligned} \quad (6.1.17)$$

ce qui montre que T est compatible avec la composition. La transformation T est ainsi un foncteur. \square

Proposition 6.1.6. *Soient M et N deux monoïdes et $\theta : M \rightarrow N$ un morphisme de monoïde. Si θ est injectif (resp. surjectif), alors $T\theta$ est injectif (resp. surjectif).*

Démonstration. Supposons θ injectif. S'il existe deux éléments x et y de TM tels que $T\theta(x) = T\theta(y)$, alors,

$$T\theta(x) = (\theta(x_1), \dots, \theta(x_n)) = (\theta(y_1), \dots, \theta(y_n)) = T\theta(y), \quad (6.1.18)$$

ce qui implique $\theta(x_i) = \theta(y_i)$ pour tout $i \in [n]$. Comme θ est injectif, on a $x_i = y_i$ pour tout $i \in [n]$, ce qui implique $x = y$. Le morphisme d'opérade $T\theta$ est donc injectif.

Supposons θ surjectif. Soit y un élément de $TN(n)$. Comme θ est surjectif, il existe des éléments x_i de M tels que $\theta(x_i) = y_i$ pour tout $i \in [n]$. Nous avons donc

$$T\theta(x_1, \dots, x_n) = (\theta(x_1), \dots, \theta(x_n)) = (y_1, \dots, y_n), \quad (6.1.19)$$

et comme (x_1, \dots, x_n) est par définition un élément de TM , le morphisme d'opérade $T\theta$ est surjectif. \square

6.2 Quelques opérades issues de la construction

Nous allons utiliser la construction décrite dans le paragraphe précédent pour définir plusieurs opérades TM associées à des monoïdes M . Ces opérades possèdent un nombre fini d'éléments d'une arité donnée si et seulement si le monoïde M est fini. Comme nous nous intéresserons par la suite principalement à des opérades ayant un nombre fini d'éléments d'une arité donnée, nous allons considérer des sous-opérades de TM engendrées par un nombre fini de générateurs.

Énonçons tout d'abord le lemme suivant qui nous sera très utile dans ce paragraphe :

Lemme 6.2.1. *Soit \mathcal{P} une opérade ns ensembliste qui est engendrée par un ensemble G de générateurs d'arités supérieures à deux. Alors, tout élément x de \mathcal{P} différent de l'unité s'écrit sous la forme $x = y \circ_i g$ où $y \in \mathcal{P}(n)$, $g \in G$ et $i \in [n]$.*

Démonstration. Nous procédons par récurrence sur l'arité de x . Si x est d'arité 2, alors $x \in G$ et x admet l'écriture $x = \mathbf{1} \circ_1 x$ où $\mathbf{1}$ est l'unité de \mathcal{P} .

Sinon, x est d'arité supérieure à 3, et, par définition d'une opérade ns ensembliste engendrée par un ensemble, x s'écrit sous la forme $x = y \circ_j z$ où $y, z \in \mathcal{P}$, $y \neq \mathbf{1}$, $z \neq \mathbf{1}$, et, comme les éléments de G sont d'arités supérieures à 2, les arités de y et de z sont strictement inférieures à l'arité de x . Par hypothèse de récurrence, il existe des éléments $y', z' \in \mathcal{P}$, des générateurs $g, h \in G$ et des entiers k, ℓ tels que $y = y' \circ_k g$ et $z = z' \circ_\ell h$. Nous avons donc

$$x = (y' \circ_k g) \circ_j (z' \circ_\ell h), \quad (6.2.1)$$

ce qui se réécrit, à l'aide de la relation d'associativité en série des opérades, en

$$x = ((y' \circ_k g) \circ_j z') \circ_{j+\ell-1} h, \quad (6.2.2)$$

et termine la démonstration du lemme. \square

Pour le reste de ce paragraphe, \mathbb{N} désigne le monoïde additif des entiers naturels et \mathbb{N}_ℓ désigne pour tout entier $\ell \geq 1$ le monoïde quotient de \mathbb{N} constitué de l'ensemble $\{0, 1, \dots, \ell - 1\}$ où le produit est l'addition modulo ℓ . Nous considérons ici des sous-opérades symétriques ou non symétriques des opérades \mathbf{TN} et \mathbf{TN}_ℓ . Remarquons au passage que le fait que \mathbf{T} soit un foncteur (théorème 6.1.5) qui respecte les surjections (proposition 6.1.6) implique que \mathbf{TN}_ℓ est une opérade quotient de \mathbf{TN} .

Les opérades définies dans ce paragraphe s'agencent selon le diagramme de la figure 6.1.

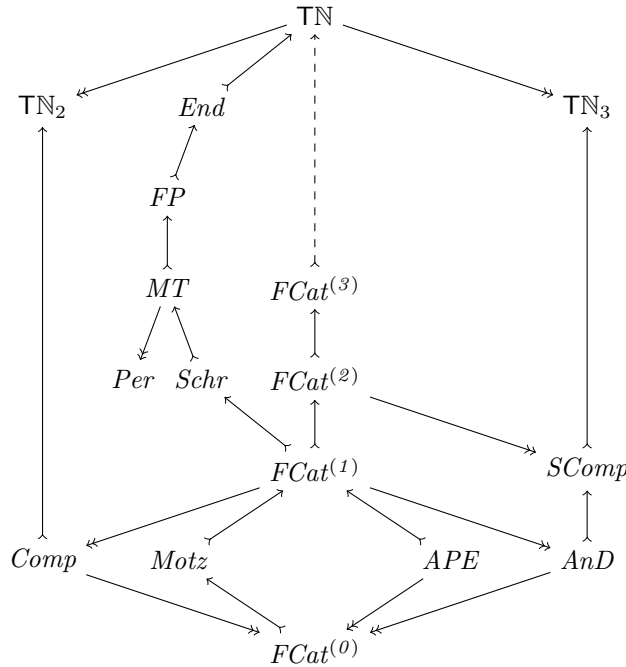


FIGURE 6.1 – Le diagramme des sous-opérades ns et quotients de \mathbf{TN} . Les flèches \hookrightarrow (resp. \twoheadrightarrow) sont des morphismes injectifs (resp. surjectifs) d'opérades ns.

6.2.1 Une opérade sur les endofonctions

Rappelons qu'une endofonction est une application de la forme $f : E \rightarrow E$ où E est un ensemble quelconque. Dans notre contexte, E est fini et nous utiliserons la définition suivante :

Définition 6.2.2. Un mot x sur l'alphabet \mathbb{N} est une endofonction si toute lettre x_i de x vérifie $0 \leq x_i \leq |x| - 1$. La taille d'une endofonction est sa longueur. L'ensemble des endofonctions de taille n est noté \mathcal{F}_n .

On retrouve les endofonctions dans leur définition habituelle par la transformation bijective qui consiste à incrémenter chaque lettre du mot. En effet, si $x \in \mathcal{F}(n)$, l'endofonction $f : [n] \rightarrow [n]$ encodée par x est définie par $f(i) := x_i + 1$. Le décalage induit par cet encodage est néanmoins nécessaire dans notre contexte puisque l'ensemble des endofonctions dans leur définition originelle n'est pas stable pour la substitution partielle de \mathbb{TN} . En effet, nous avons par exemple

$$12 \circ_2 12 = 134, \quad (6.2.3)$$

et, bien que 12 soit une endofonction dans sa définition habituelle, 134 n'en est pas une.

Proposition 6.2.3. L'ensemble $End := \biguplus_{n \geq 1} \mathcal{F}_n$ forme une sous-opérade de \mathbb{TN} .

Démonstration. Il s'agit de montrer que End est stable pour la substitution partielle de \mathbb{TN} . Soient $x \in End(n)$, $y \in End(m)$ et $i \in [n]$. Notons qu'un mot z est une endofonction si et seulement si $0 \leq \min z$ et $\max z \leq |z| - 1$. Comme l'opération de substitution ne décrémente pas les valeurs des lettres des opérandes, on a $0 \leq \min x \circ_i y$. De plus,

$$\max x \circ_i y = \max\{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n\} \cup \{x_i + y_1, \dots, x_i + y_m\} \quad (6.2.4)$$

$$\leq \max\{|x| - 1\} \cup \{|x| - 1 + |y| - 1\} \quad (6.2.5)$$

$$\leq |x| + |y| - 2, \quad (6.2.6)$$

ce qui montre que $x \circ_i y$, de longueur $|x| + |y| - 1$, est bien une endofonction.

L'unité de \mathbb{TN} est toujours un élément de End . En effet, c'est l'élément $\mathbf{1} = (0)$, qui est bien une endofonction.

Pour finir, End est stable pour l'action du groupe symétrique car si x est une endofonction, toute permutation de x l'est également. \square

Voici quelques exemples de substitutions partielles dans End :

$$110 \circ_3 11 = 1111, \quad (6.2.7)$$

$$212 \circ_2 3013 = 241242, \quad (6.2.8)$$

et d'applications de l'action du groupe symétrique :

$$11210 \cdot 23514 = 12011, \quad (6.2.9)$$

$$341413 \cdot 361425 = 133441. \quad (6.2.10)$$

Cette opérade n'est pas finiment engendrée. En effet, pour tout $n \geq 2$, les éléments $x \in End(n)$ vérifiant $x_i = n - 1$ pour tout $i \in [n]$ ne peuvent pas être obtenus par des substitution partielles d'éléments d'arités strictement inférieures.

D'après la proposition 6.2.3, nous avons $\dim End(n) = n^n$. La suite des dimensions de End forme la suite **A000312** de [Slo] et ses premières valeurs sont

$$1, 4, 27, 256, 3125, 46656, 823543. \quad (6.2.11)$$

6.2.2 Une opérade sur les fonctions de parking

Nous utiliserons, dans notre contexte, la définition suivante des fonctions de parking :

Définition 6.2.4. *Un mot x sur l'alphabet \mathbb{N} est une fonction de parking si pour toute lettre x_i de x , on a $0 \leq x_i \leq |x| - 1$ et il existe un sous-mot y de x de longueur x_i tel que toutes les lettres de y sont strictement inférieures à x_i . La taille d'une fonction de parking est sa longueur. L'ensemble des fonctions de parking de taille n est noté $FP(n)$.*

On retrouve les fonctions de parking dans leur définition habituelle par la transformation bijective visant à incrémenter chaque lettre du mot. Ce choix est motivé par les mêmes raisons qui nous ont poussé à considérer une définition alternative des endofonctions dans le paragraphe 6.2.1.

Proposition 6.2.5. *L'ensemble $FP := \biguplus_{n \geq 1} FP(n)$ forme une sous-opérade de End .*

Démonstration. Nous avons, immédiatement depuis la définition 6.2.4, $FP(n) \subseteq End(n)$ pour tout $n \geq 1$. Montrons que FP est stable pour la substitution partielle. Soient $x \in FP(n)$, $y \in FP(m)$, $i \in [n]$ et $z := x \circ_i y$. Montrons que z est une fonction de parking. Pour cela, montrons que pour toute lettre z_j de z , il existe un sous-mot u de z de longueur z_j constitué de lettres strictement inférieures à z_j . Nous avons deux cas à explorer en fonction de la valeur de j .

Si $1 \leq j \leq i - 1$ ou $i + m \leq j \leq n + m - 1$, alors z_j est aussi une lettre de x . Comme x est une fonction de parking, il existe un sous-mot v de x de longueur z_j dont les lettres sont strictement inférieures à z_j . Maintenant, si v n'a aucune lettre en commun avec la lettre x_i , alors le mot $u := v$ convient. En revanche, si v est formé de la lettre x_i , comme y est une fonction de parking, il admet une occurrence de la lettre 0, si bien que v est encore un sous-mot de z , et de ce fait, le mot $u := v$ convient.

Sinon, lorsque $i \leq j \leq i + m - 1$, la lettre z_j se trouve dans le facteur de la forme $w := (x_i + y_1, \dots, x_i + y_m)$ de z . Dans ce cas, comme y est une fonction de parking et que z admet comme facteur le mot y dont les lettres sont incrémentées de x_i , il existe un sous-mot v de w de longueur $z_j - x_i$ et tel que ses lettres sont strictement inférieures à $z_j - x_i$. De plus, comme x est aussi une fonction de parking, il existe un sous-mot v' de x de longueur x_i composé de lettres strictement inférieures à x_i . Ce sous-mot est bien entendu aussi un sous-mot de z . Par conséquent, un mélange adéquat de v et v' produit un sous-mot u de z de longueur $z_j - x_i + x_i = z_j$ et tel que toutes ses lettres sont bien strictement inférieures à z_j .

L'unité de End est toujours un élément de FP . En effet, c'est l'élément $\mathbf{1} = (0)$, qui est bien une fonction de parking.

Pour finir, FP est stable pour l'action du groupe symétrique car si x est une fonction de parking, toute permutation de x l'est par définition également. \square

Voici un exemple de substitution partielle dans FP :

$$0202 \circ_2 1013 = 0323502. \quad (6.2.12)$$

Cette opérade n'est pas finiment engendrée car, pour tout $n \geq 3$, les éléments $x \in FP(n)$ vérifiant $x_i = 0$ pour tout $i \in [n - 1]$ et $x_n = n - 1$ ne peuvent pas être obtenus par des substitutions partielles d'éléments d'arités strictement inférieures.

D'après la proposition 6.2.5, nous avons $\dim FP(n) = (n + 1)^{(n-1)}$. La suite des dimensions de FP forme la suite **A000272** de [Slo] et ses premières valeurs sont

$$1, 3, 16, 125, 1296, 16807, 262144. \quad (6.2.13)$$

6.2.3 Une opérade sur les mots tassés

Nous utiliserons, dans notre contexte, la définition suivante des mots tassés :

Définition 6.2.6. *Un mot x sur l'alphabet \mathbb{N} est un mot tassé si pour toute lettre x_i de x on a $0 \leq x_i \leq |x| - 1$ et de plus, si $x_i \geq 1$, alors $x_i - 1$ apparaît dans x . La taille d'un mot tassé est sa longueur. L'ensemble des mots tassés de taille n est noté $MT(n)$.*

Notons que l'on retrouve les mots tassés dans leur définition habituelle par la transformation bijective qui consiste à incrémenter chaque lettre du mot. Ce choix est motivé par les mêmes raisons qui nous ont poussé à considérer une définition alternative des endofonctions et des fonctions de parking dans les paragraphes 6.2.1 et 6.2.2.

Proposition 6.2.7. *L'ensemble $MT := \biguplus_{n \geq 1} MT(n)$ forme une sous-opérade de FP .*

Démonstration. Nous avons, immédiatement depuis la définition 6.2.6, $MT(n) \subseteq FP(n)$ pour tout $n \geq 1$. Montrons que MT est stable pour la substitution partielle. Soient $x \in MT(n)$, $y \in MT(m)$, et $i \in [n]$. Montrons que $z := x \circ_i y$ est un mot tassé. Pour cela, montrons que pour toute lettre z_j de z vérifiant $z_j \geq 1$, il existe une lettre a de z telle que $a = z_j - 1$. Nous avons deux cas à explorer en fonction de la valeur de j .

Si $1 \leq j \leq i - 1$ ou $i + m \leq j \leq n + m - 1$, la lettre z_j est aussi dans x . Comme x est un mot tassé, il existe un indice k tel que $x_k = z_j - 1$. Maintenant, si $k \neq i$, la lettre $a := x_k = z_j - 1$ se trouve également dans le mot z . Sinon, lorsque $k = i$, comme y est un mot tassé, il possède au moins une occurrence de la lettre 0, ce qui implique l'existence d'une lettre de valeur $a := x_i = x_k = z_j - 1$ dans z .

Sinon, lorsque $i \leq j \leq i + m - 1$, la lettre z_j se trouve dans le facteur de la forme $(x_i + y_1, \dots, x_i + y_m)$ de z et on a $z_j = x_i + y_k$ où y_k est une lettre de y . Maintenant, si $y_k \geq 1$, comme y est un mot tassé, il existe une lettre y_ℓ dans y telle que $y_\ell = y_k - 1$. Il existe donc une lettre de valeur $a := x_i + y_\ell = x_i + y_k - 1 = z_j - 1$ dans z . Sinon, lorsque $y_k = 0$, on a $z_j = x_i$, et comme x est un mot tassé, il existe une lettre x_ℓ dans x telle que $a := x_\ell = x_i - 1 = z_j - 1$. Cette lettre a existe également dans z .

L'unité de FP est toujours un élément de MT . En effet, c'est l'élément $\mathbf{1} = (0)$, qui est bien un mot tassé.

Pour finir, MT est stable pour l'action du groupe symétrique car si x est un mot tassé, toute permutation de x l'est par définition également. \square

Contrairement aux opérades End et FP , l'opérade MT est finiment engendrée :

Proposition 6.2.8. *L'opérade MT est la sous-opérade de $\mathbb{T}\mathbb{N}$ engendrée, en tant qu'opérade symétrique, par les générateurs 00 et 01.*

Démonstration. Tout d'abord, d'après la proposition 6.2.7, étant donné que 00 et 01 sont des mots tassés, les éléments de $\langle 00, 01 \rangle$ sont aussi des mots tassés.

Soit x un mot tassé croissant. Montrons par récurrence sur la longueur de x que $x \in \langle 00, 01 \rangle$. Si $|x| = 1$, alors, comme x est un mot tassé, la seule possibilité est $x = 0 = \mathbf{1}$ et comme $\mathbf{1} \in \langle 00, 01 \rangle$, le cas de base est vérifié. Sinon, considérons un mot tassé croissant y de longueur $n+1$ avec $n \geq 1$. Posons $x := y_1 \dots y_n$ le préfixe de longueur n de y . Comme x est de longueur n , par hypothèse de récurrence, $x \in \langle 00, 01 \rangle$. Maintenant, étant donné que y est croissant et est un mot tassé, il existe exactement deux possibilités pour exprimer sa dernière lettre en fonction de son avant-dernière : si $y_{n+1} = y_n$, alors on a $y = x \circ_n 00$. Si $y_{n+1} = y_n + 1$, alors on a $y = x \circ_n 01$. Ceci montre que tout mot tassé croissant y vérifie $y \in \langle 00, 01 \rangle$. De plus, tout mot tassé x peut être obtenu à partir d'un mot tassé croissant y par un réordonnement de ses lettres. En d'autres termes, on a $x = y \cdot \sigma$ pour une certaine permutation σ . Cela implique le résultat. \square

D'après la proposition 6.2.7, nous avons $\dim MT(n) = \sum_{i \in [n]} i! \operatorname{stir}(n, i)$ où $\operatorname{stir}(n, i)$ dénote le nombre de Stirling de seconde espèce en n et i , c'est-à-dire le nombre de partitions en i sous-ensembles d'un ensemble à n éléments. La suite des dimensions de MT forme la suite **A000670** de [Slo] et ses premières valeurs sont

$$1, 3, 13, 75, 541, 4683, 47293. \quad (6.2.14)$$

6.2.4 Une opérade sur les permutations

Un simple essai suffit à se rendre compte que l'ensemble des permutations n'est pas stable dans \mathbf{TN} . Nous allons cependant construire dans ce paragraphe une opérade sur les permutations comme un quotient de MT . Nous considérons également dans ce paragraphe l'opérade ensembliste \mathbf{TN} et ses sous-opérades comme des opérades non ensemblistes, où la substitution partielle et l'action du groupe symétrique sont étendues par linéarité.

Soit $I := \bigoplus_{n \geq 1} I(n)$ le sous-espace vectoriel de MT où, pour tout $n \geq 1$,

$$I(n) := \operatorname{Vect}(MT(n) \setminus \mathfrak{S}_n). \quad (6.2.15)$$

L'espace vectoriel I est donc engendré par les mots tassés qui possèdent au moins une double occurrence d'une lettre.

Proposition 6.2.9. *L'espace vectoriel I un idéal d'opérade de MT .*

Démonstration. Soient $x \in I(n)$, $y \in MT(m)$ et $i \in [n]$. Montrons que $z := x \circ_i y \in I(n+m-1)$. Notons d'abord que le mot z est bien un mot tassé puisque $I(n) \subseteq MT(n)$. Comme x est un mot tassé mais pas une permutation, il existe deux indices $j < k$ tels que $x_j = x_k$. Si $i \neq j$ et $i \neq k$, le mot z est de la forme $z = u \cdot x_j \cdot v \cdot x_k \cdot w$ avec $u, v, w \in \mathbf{TN}$. Comme z possède deux occurrences d'une même lettre, ce n'est pas une permutation. On a donc $z \in I$. Si $i = j$ ou $i = k$, comme y est un mot tassé, il possède une occurrence de la lettre 0. Lors de son insertion dans x , cette lettre 0 est changée en la lettre x_i . Le mot z possède donc deux occurrences d'une même lettre, ce qui montre que l'on a également $z \in I$ dans ce cas.

Soient $x \in MT(n)$, $y \in I(m)$ et $i \in [n]$. Montrons que $z := x \circ_i y \in I(n+m-1)$. Notons d'abord que le mot z est bien un mot tassé puisque $I(n) \subseteq MT(n)$. Comme y est un mot tassé mais pas une permutation, il existe deux indices $j < k$ tels que $y_j = y_k$. Il est de la forme $y = u \cdot y_j \cdot v \cdot y_k \cdot w$ avec $u, v, w \in \mathbf{TN}$. Le mot z est obtenu en insérant dans x une version de y dont les lettres ont été incrémentées de la valeur de x_i . Le mot z possède donc les deux occurrences identiques $y_j + x_i$ et $y_k + x_i$, montrant que z n'est pas une permutation et ainsi, $z \in I$.

Il reste à montrer que pour tout $n \geq 1$, $I(n)$ est invariant pour l'action du groupe symétrique. C'est le cas car si $x \in I(n)$, alors pour toute permutation $\sigma \in \mathfrak{S}_n$, le mot $x \cdot \sigma$ possède la même évaluation que x . C'est donc un mot tassé, mais pas une permutation. \square

Nous déduisons de la proposition 6.2.9 que la structure $Per := MT/I$ est une opérade. Puisque l'espace vectoriel MT/I contient uniquement les mots tassés qui ne possèdent pas d'occurrence multiple de lettre, les bases de $Per(n)$ sont indexées par les permutations de l'ensemble $\{0, \dots, n-1\}$.

Soient $n, m \geq 2$, $x \in Per(n)$, $y \in Per(m)$ et $i \in [n]$. Si $x_i \neq n-1$, il est clair que l'élément $x \circ_i y$ possède une double occurrence de la lettre $x_i - 1$. Autrement, si $x_i = n-1$, l'élément $x \circ_i y$ est une permutation. Ces observations mènent à l'expression suivante de la substitution partielle dans Per :

$$x \circ_i y = \begin{cases} x \circ_i y & \text{si } x_i = \max x, \\ 0_{\mathbb{K}} & \text{sinon,} \end{cases} \quad (6.2.16)$$

où la substitution partielle dans le membre droit de (6.2.16) est celle de l'opérade MT .

Par la proposition 6.2.8, MT est finiment engendrée, en tant qu'opérade symétrique, par les éléments 00 et 01. La définition de Per implique donc que cette dernière est également finiment engendrée, en tant qu'opérade symétrique, par l'élément 01.

Voici quelques exemples de substitutions partielles dans Per :

$$\mathbf{20431} \circ_1 \mathbf{102} = \mathbf{0_{\mathbb{K}}}, \quad (6.2.17)$$

$$\mathbf{20431} \circ_3 \mathbf{102} = \mathbf{2054631}. \quad (6.2.18)$$

6.2.5 Une opérade sur les arbres plans enracinés

Considérons la sous-opérade ns APE de TN engendrée par l'élément 01. Voici les premiers éléments qu'elle contient, arité par arité :

- arité 1 : 0 ;
- arité 2 : 01 ;
- arité 3 : 011, 012 ;
- arité 4 : 0111, 0112, 0121, 0122, 0123 ;
- arité 5 : 01111, 01112, 01121, 01122, 01123, 01211, 01212, 01221, 01222, 01223, 01231, 01232, 01233, 01234.

Les éléments de APE sont caractérisés de la manière suivante :

Lemme 6.2.10. *Les éléments de l'opérade ns APE sont exactement les mots x sur l'alphabet \mathbb{N} qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.*

Démonstration. Commençons par montrer par récurrence sur la longueur des mots que tout mot y engendré par APE vérifie les conditions du lemme. Tout d'abord, si $|y| = 1$, alors la seule possibilité est $y = \mathbf{1} = 0$ et y vérifie les conditions du lemme. Sinon, d'après le lemme 6.2.1, il existe un élément x de APE de longueur n et $i \in [n]$ tels que $y := x \circ_i 01$. Nous avons alors

$$y = (x_1, \dots, x_{i-1}, x_i, x_i + 1, x_{i+1}, \dots, x_n). \quad (6.2.19)$$

Étant donné que, par hypothèse de récurrence, x vérifie les conditions du lemme, que $y_1 = x_1 = 0$, que

$$y_{i+1} = x_i + 1 \leq x_i + 1 = y_i + 1, \quad (6.2.20)$$

et que

$$y_{i+2} = x_{i+1} \leq x_i + 1 = y_{i+1} \leq y_{i+1} + 1, \quad (6.2.21)$$

le mot y vérifie les conditions du lemme.

Montrons maintenant par récurrence sur la longueur des mots que APE permet d'engendrer tous les mots qui vérifient les conditions du lemme. C'est bien le cas pour le mot $0 = \mathbf{1}$ de longueur 1. Considérons maintenant un mot x de longueur $n \geq 2$ et qui vérifie les conditions du lemme. De part le fait que l'on a toujours $x_1 = 0$ et $x_2 = 1$, il existe dans x un facteur de la forme $x_i \cdot x_{i+1}$ où i est le plus grand indice tel que $x_{i+1} = x_i + 1$. De cela, en posant $y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n)$, on en déduit que $x = y \circ_i 01$. Maintenant, comme i est maximal, on a soit $i + 1 = n$, soit $x_{i+2} \leq x_{i+1}$. Ceci implique que le mot y respecte les conditions du lemme. Par hypothèse de récurrence, y est engendré par APE , et comme $x = y \circ_i 01$, x l'est également. \square

Si T est un arbre plan enraciné, on appelle i^e *nœud* de T le i^e nœud que l'on visite lors d'un parcours en profondeur gauche droite de T .

D'après la caractérisation des éléments de APE fournie par le lemme 6.2.10, on peut établir une bijection, pour tout $n \geq 1$, entre les éléments de $APE(n)$ et les arbres plans enracinés à n nœuds de la manière suivante. Soit $x \in APE$. On traite les lettres de x de la gauche vers la

droite en créant un nœud pour chacune de ses lettres : chaque lettre x_i correspond au i^{e} nœud de l'arbre construit. La lettre $x_1 = 0$ correspond à la racine, et ensuite, à la lecture d'une lettre x_i , on crée un nœud de profondeur x_i , fils le plus à droite possible du dernier nœud créé de profondeur x_{i-1} . La condition $x_1 = 0$ garantit que le premier nœud créé est bien de profondeur zéro et donc une racine, et la condition $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$ garantit à la fois qu'aucun autre nœud de profondeur zéro n'est créé et que tout nœud possède un parent.

L'inverse de cette bijection s'obtient simplement en réalisant un parcours en profondeur gauche droite de l'arbre plan enraciné et en notant successivement les profondeurs des nœuds visités. La figure 6.2 illustre un exemple de cette bijection.

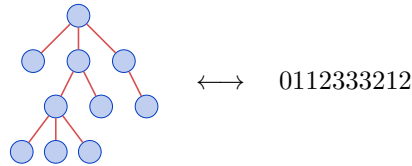


FIGURE 6.2 – Un arbre plan enraciné et l'élément correspondant de l'opérade ns APE .

Cette bijection et le fait que les arbres plans enracinés à $n + 1$ nœuds sont énumérés selon la suite de Catalan $\frac{1}{n+1} \binom{2n}{n}$ (voir [Sta99a]) impliquent que la série de Hilbert de APE vérifie

$$F_{APE}(t) = \frac{1 - \sqrt{1 - 4t}}{2} = \sum_{n \geq 1} \frac{1}{n} \binom{2n-2}{n-1} t^n \quad (6.2.22)$$

$$= t + t^2 + 2t^3 + 5t^4 + 14t^5 + 42t^6 + 132t^7 + 429t^8 + 1430t^9 + \dots \quad (6.2.23)$$

Cette bijection peut également être utilisée pour donner une description de la substitution partielle de APE en termes d'arbres plans enracinés :

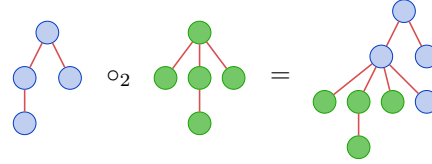
Lemme 6.2.11. *Soient T et S deux arbres plans enracinés et t le i^{e} nœud de T . La substitution partielle $T \circ_i S$ dans l'opérade ns APE revient à greffer dans T les sous-arbres de la racine de S comme fils les plus à gauche de t .*

Démonstration. Soit $x \in APE(n)$ (resp. $y \in APE(m)$) l'élément correspondant à T (resp. S) selon la bijection que nous venons de décrire. Nous avons alors

$$z := x \circ_i y = (x_1, \dots, x_{i-1}, x_i + y_1, \dots, x_i + y_m, x_{i+1}, \dots, x_n). \quad (6.2.24)$$

L'arbre en bijection avec le préfixe $(x_1, \dots, x_{i-1}, x_i + y_1)$ de z est, d'après la définition de la bijection et le fait que $y_1 = 0$, l'arbre T restreint à ses i premiers nœuds. Ensuite, encore d'après la bijection, l'arbre correspondant au préfixe $(x_1, \dots, x_{i-1}, x_i + y_1, \dots, x_i + y_m)$ est l'arbre T restreint à ses i premiers nœuds, auquel sont greffés sur son i^{e} nœud t les sous-arbres de la racine de S . Remarquons que les nœuds qui proviennent de S sont, une fois greffés, de profondeur strictement supérieure à x_i . Finalement, d'après le lemme 6.2.10, nous avons $1 \leq x_{i+1} \leq x_i + 1$, impliquant $1 \leq x_{i+1} \leq x_i + y_j$ pour tout $2 \leq j \leq m$. Ainsi, les $n - i$ derniers nœuds qui proviennent de T se retrouvent greffés à des nœuds de profondeur inférieure à x_i , qui sont précisément les i premiers nœuds de T . Les $n - i$ derniers nœuds se retrouvent ainsi greffés de la même manière qu'ils l'étaient dans T . Ceci implique que l'arbre en bijection avec z est conforme à la description du lemme. \square

La figure 6.3 montre un exemple de substitution dans APE en se basant sur l'interprétation fournie par le lemme 6.2.11.

FIGURE 6.3 – Un exemple de substitution partielle dans l'opérade ns APE .

Proposition 6.2.12. *L'opérade ns APE est isomorphe à l'opérade ns libre sur un générateur d'arité deux.*

Démonstration. En se basant ce qui est exposé dans le paragraphe 3.1.2 du chapitre 3, les éléments d'arité n de l'opérade ns libre sur un générateur d'arité deux peuvent s'identifier à des arbres binaires à n feuilles. La conclusion tient au fait qu'il n'y a pas de relation quadratique mettant en jeu le générateur 01 de APE et qu'il ne peut y en avoir non plus de plus haut degré car la série de Hilbert (6.2.22) de APE est la même que la série de Hilbert de l'opérade ns libre sur un générateur d'arité deux. \square

L'opérade ns APE peut se voir comme une version planaire de l'opérade NAP .

6.2.6 Une famille d'opérades sur les k -chemins de Dyck

Soit $k \geq 0$ un entier et $FCat^{(k)}$ la sous-opérade ns de TN engendrée par l'ensemble

$$\{00, 01, \dots, 0k\}. \quad (6.2.25)$$

Voici par exemple les premiers éléments de $FCat^{(1)}$, arité par arité :

- arité 1 : 0 ;
 - arité 2 : 00, 01 ;
 - arité 3 : 000, 001, 010, 011, 012 ;
 - arité 4 : 0000, 0001, 0010, 0011, 0012, 0100, 0101, 0110, 0111, 0112, 0120, 0121, 0122, 0123 ;
- et ceux de $FCat^{(2)}$:

- arité 1 : 0 ;
- arité 2 : 00, 01, 02 ;
- arité 3 : 000, 001, 002, 010, 011, 012, 013, 020, 021, 022, 023, 024.

Les éléments de $FCat^{(k)}$ sont caractérisés de la manière suivante :

Lemme 6.2.13. *Les éléments de l'opérade ns $FCat^{(k)}$ sont exactement les mots x sur l'alphabet \mathbb{N} qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.*

Démonstration. Commençons par montrer par récurrence sur la longueur des mots que tout élément y engendré par $FCat^{(k)}$ vérifie les conditions du lemme. Si $|y| = 1$, alors $y = \mathbf{1} = 0$ et y vérifie les conditions du lemme. Sinon, d'après le lemme 6.2.1, il existe un élément x de $FCat^{(k)}$ de longueur n , $i \in [n]$ et $0 \leq h \leq k$ tels que $y := x \circ_i 0h$. Nous avons alors

$$y = (x_1, \dots, x_{i-1}, x_i, x_i + h, x_{i+1}, \dots, x_n). \quad (6.2.26)$$

Étant donné que, par hypothèse de récurrence, x vérifie les conditions du lemme, que $y_1 = x_1 = 0$, que

$$y_{i+1} = x_i + h \leq x_i + k = y_i + k, \quad (6.2.27)$$

et que

$$y_{i+2} = x_{i+1} \leq x_i + k \leq x_i + h + k = y_{i+1} + k, \quad (6.2.28)$$

le mot y vérifie les conditions du lemme.

Montrons maintenant par récurrence sur la longueur des mots que $FCat^{(k)}$ permet d'engendrer tous les mots qui vérifient les conditions du lemme. C'est bien le cas pour le mot 0 de longueur un. Considérons maintenant un mot x de longueur $n \geq 2$ qui vérifie les conditions du lemme. Soit $i \in [n-1]$ le plus grand indice tel que $x_i \leq x_{i+1}$. Notons que cet indice existe nécessairement puisque $x_1 = 0$ et $0 \leq x_2 \leq k$. De cela, en posant $h := x_{i+1} - x_i$ et $y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n)$, nous avons $x = y \circ_i 0h$. Maintenant, si $i = |y|$, le mot y respecte les conditions du lemme puisque y est un préfixe de x . Sinon, lorsque $i \leq |y| - 1$, comme i est maximal, on a $x_{i+2} + 1 \leq x_{i+1}$, et ceci implique que y respecte également dans ce cas les conditions du lemme. Dans ces deux cas, par hypothèse de récurrence, y est engendré par $FCat^{(k)}$ et comme $x = y \circ_i 0h$, x l'est également. \square

Nous avons maintenant besoin de la définition des k -chemins de Dyck, que nous rappelons ici.

Définition 6.2.14. Un k -chemin de Dyck de taille n est un chemin dans le réseau \mathbb{Z}^2 qui relie les points de coordonnées $(0, 0)$ et $((k+1)n, 0)$ par n pas montants $(1, k)$ et kn pas descendants $(1, -1)$, avec la condition supplémentaire que le chemin reste au dessus de la droite horizontale d'ordonnée 0.

Il est bien connu [DM47] que les k -chemins de Dyck sont dénombrés par la suite de Fuss-Catalan, dont le n^e terme est donné par

$$\frac{1}{kn+1} \binom{(k+1)n}{n}. \quad (6.2.29)$$

Notons que l'on retrouve les chemins de Dyck classiques, et ainsi les nombres de Catalan, avec la spécialisation $k := 1$.

Il existe une bijection h , justifiée par le lemme 6.2.13, entre les k -chemins de Dyck et les éléments de l'opérade $FCat^{(k)}$. Celle-ci possède la description suivante. Si c est un k -chemin de Dyck de taille n , on construit le mot $h(c)$ en notant de gauche à droite les ordonnées des points de départ de chacun des pas montants de c . Le mot $h(c)$ respecte les conditions du lemme 6.2.13 et est de ce fait un élément de $FCat^{(k)}$. La figure 6.4 illustre un exemple de cette bijection.

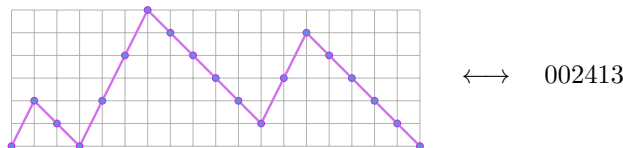


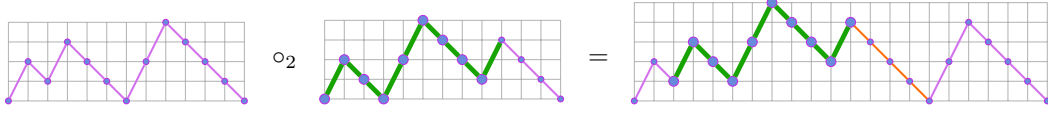
FIGURE 6.4 – Un 2-chemin de Dyck et l'élément correspondant de l'opérade $FCat^{(2)}$.

On déduit de cette bijection que la série de Hilbert de $FCat^{(k)}$ est

$$F_{FCat^{(k)}}(t) = \sum_{n \geq 1} \frac{1}{kn+1} \binom{(k+1)n}{n} t^n. \quad (6.2.30)$$

Nous pouvons de plus reformuler la substitution partielle dans $FCat^{(k)}$ en termes de k -chemins de Dyck de la manière suivante. Soient c et d deux k -chemins de Dyck. La substitution partielle $c \circ_i d$ revient à supprimer le i^e pas montant de c ainsi que les pas descendants qui le suivent, à placer à cet endroit le préfixe de d privé de son plus long suffixe constitué uniquement de pas descendants, et à recoller la partie restante de droite de c en complétant si nécessaire avec des pas descendants au milieu. La figure 6.5 montre un exemple de substitution.

Nous notons à présent par α_i un générateur $0i$ de $FCat^{(k)}$ pour tout $0 \leq i \leq k$.

FIGURE 6.5 – Un exemple de substitution partielle dans l'opérade ns $FCat^{(2)}$.

Proposition 6.2.15. *Les générateurs $\{\alpha_0, \dots, \alpha_k\}$ de l'opérade ns $FCat^{(k)}$ sont soumis, en degré deux, uniquement aux relations*

$$\alpha_i \circ_1 \alpha_j = \alpha_j \circ_2 \alpha_{i-j}, \quad (6.2.31)$$

pour tous $0 \leq j \leq i \leq k$.

Démonstration. La relation (6.2.31) est bien vérifiée : nous avons en effet

$$\alpha_i \circ_1 \alpha_j = (0, j, i), \quad (6.2.32)$$

et

$$\alpha_j \circ_2 \alpha_{i-j} = (0, j, i - j + j) = (0, j, i). \quad (6.2.33)$$

Une simple inspection montre que ce sont les seules relations de degré deux. En effet, les relations $\alpha_i \circ_1 \alpha_j = \alpha_k \circ_1 \alpha_\ell$ et $\alpha_i \circ_2 \alpha_j = \alpha_k \circ_2 \alpha_\ell$ impliquent $i = k$ et $j = \ell$. \square

Remarquons que la proposition 6.2.15 ne signifie pas que les seules relations non triviales de $FCat^{(k)}$ sont de la forme (6.2.31).

Notons que la relation $\alpha_0 \circ_1 \alpha_0 = \alpha_0 \circ_2 \alpha_0$ entraîne simplement l'associativité de α_0 . La relation $\alpha_1 \circ_1 \alpha_1 = \alpha_1 \circ_2 \alpha_0$ est l'une de celles qui figurent dans l'opérade diptère [LR03], [Zin10] et la relation $\alpha_1 \circ_1 \alpha_0 = \alpha_0 \circ_2 \alpha_1$ figure quant à elle dans l'opérade dupliciale [Lod08], [Zin10].

6.2.7 Une opérade sur les arbres de Schröder

Considérons maintenant l'opérade ns $Schr$, sous-opérade ns de TN engendrée par l'ensemble $\{00, 01, 10\}$. Listons ses premiers éléments, arité par arité :

- arité 1 : 0 ;
- arité 2 : 00, 01, 10 ;
- arité 3 : 000, 001, 010, 011, 012, 021, 100, 101, 110, 120, 210 ;
- arité 4 : 0000, 0001, 0010, 0011, 0012, 0021, 0100, 0101, 0110, 0111, 0112, 0120, 0121, 0122, 0123, 0132, 0210, 0211, 0212, 0221, 0231, 0321, 1000, 1001, 1010, 1011, 1012, 1021, 1100, 1101, 1110, 1120, 1200, 1201, 1210, 1220, 1230, 1320, 2100, 2101, 2110, 2120, 2210, 2310, 3210.

Les éléments de $Schr$ sont caractérisés de la manière suivante :

Lemme 6.2.16. *Les éléments de l'opérade ns $Schr$ sont exactement les mots x sur l'alphabet \mathbb{N} , qui possèdent au moins une occurrence de la lettre 0 et tels que pour toute lettre $x_i \geq 1$, il existe une lettre $x_j := x_i - 1$ telle que le facteur de x situé entre x_i et x_j soit composé uniquement de lettres x_k vérifiant $x_k \geq x_i$.*

Démonstration. Montrons d'abord que tout mot de $Schr$ vérifie les conditions du lemme. On procède par récurrence sur la longueur des mots. Le cas de base est vérifié car $x := 1 = 0$ respecte les conditions du lemme. Soit x un élément de $Schr$ d'arité $n \geq 2$. D'après le lemme 6.2.1, x peut s'écrire sous la forme $x = y \circ_i g$ où y est un élément d'arité $n - 1$ et $g \in \{00, 01, 10\}$. Nous avons maintenant trois cas à examiner en fonction de g . Si $g = 00$, comme

$$x = (y_1, \dots, y_{i-1}, y_i, y_i, y_{i+1}, \dots, y_{n-1}), \quad (6.2.34)$$

et que, par hypothèse de récurrence, y vérifie le lemme, c'est aussi le cas pour x . Si $g = 01$, nous avons

$$x = (y_1, \dots, y_{i-1}, y_i, y_i + 1, y_{i+1}, \dots, y_{n-1}), \quad (6.2.35)$$

et on observe facilement que x vérifie les conditions du lemme puisque y les vérifie. Le cas $g = 10$ est analogue au précédent.

Montrons maintenant que $Schr$ permet d'engendrer tous les mots qui satisfont les conditions du lemme. Nous procédons par récurrence sur la longueur des mots. Il existe un unique mot $x := 0 = \mathbf{1}$ de longueur 1 qui respecte les conditions du lemme et celui-ci figure bien dans $Schr$. Observons de plus que tout mot x constitué uniquement de 0 appartient bien à $Schr$ puisqu'il est engendré par substitutions partielles successives du générateur 00. Soit x un mot vérifiant les conditions du lemme, de longueur $n \geq 2$ et qui possède au moins une occurrence d'une lettre différente de 0. Posons $\mathbf{b} := \max x$ et $\mathbf{a} := \mathbf{b} - 1$. Il existe alors un facteur $x_i \cdot x_{i+1}$ dans x tel que $x_i \cdot x_{i+1} \in \{\mathbf{bb}, \mathbf{ba}, \mathbf{ab}\}$. En effet, si un tel facteur n'existait pas, cela entraînerait l'existence d'un facteur u non vide séparant les lettres \mathbf{b} et \mathbf{a} dans x , et étant donné que \mathbf{b} est la plus grande lettre de x , les lettres de u seraient toutes strictement inférieures à \mathbf{b} , ce qui montrerait que x ne respecte pas les conditions du lemme. Supposons dans un premier temps que x admet un facteur \mathbf{bb} . Nous avons donc

$$x = (x_1, \dots, x_i = \mathbf{b}, x_{i+1} = \mathbf{b}, \dots, x_n), \quad (6.2.36)$$

et en posant

$$y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n), \quad (6.2.37)$$

nous avons $x = y \circ_i 00$. Comme x satisfait les conditions du lemme et que y est obtenu en supprimant dans x sa plus grande lettre, y les satisfait également. L'application de l'hypothèse de récurrence sur y implique que x est un élément de $Schr$. Dans l'autre cas, si x admet un facteur \mathbf{ba} , nous avons

$$x = (x_1, \dots, x_i = \mathbf{b}, x_{i+1} = \mathbf{a}, \dots, x_n), \quad (6.2.38)$$

et en posant

$$y := (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n), \quad (6.2.39)$$

on nous avons $x = y \circ_i 10$. Pour les mêmes raisons que dans le cas précédent, le mot x est un élément de $Schr$. Le cas où x admet un facteur \mathbf{ab} est analogue au précédent. \square

Nous avons besoin maintenant de la définition des arbres Schröder, que nous rappelons ici.

Définition 6.2.17. *Un arbre de Schröder est un arbre plan enraciné tel qu'aucun nœud ne possède une unique fils. Les nœuds qui possèdent au moins deux fils sont appelés nœuds internes ou simplement nœuds et les nœuds sans fils sont appelés feuilles. La taille d'un arbre de Schröder est son nombre de feuilles.*

Décrivons une construction \mathcal{S} , analogue de celle exposée dans [NT07] et [HNT08b], qui étant donné un mot u de longueur $n \geq 1$ sur l'alphabet \mathbb{N} , construit un arbre de Schröder de taille $n + 1$. Posons $\mathbf{a} := \min u$. Le mot u admet alors l'écriture $u = u^{(1)} \cdot \mathbf{a} \cdot \dots \cdot \mathbf{a} \cdot u^{(\ell)}$ où aucun des $u^{(i)}$ ne possède d'occurrence de \mathbf{a} . De cette façon, on définit \mathcal{S} de manière récursive par

$$\mathcal{S}(u) := \begin{cases} \wedge(\perp, \perp) & \text{si } |u| = 1, \\ \wedge(\mathcal{S}(u^{(1)}), \dots, \mathcal{S}(u^{(\ell)})) & \text{sinon,} \end{cases} \quad (6.2.40)$$

où \wedge est l'opérateur qui, à une suite s d'arbres plans enracinés, associe l'arbre obtenu en créant une racine qui admet, de gauche à droite, les arbres de s comme sous-arbres (voir le paragraphe 1.3.2 du chapitre 1).

Inversement, notons \mathcal{S}^{-1} le procédé qui extrait un mot à partir d'un arbre de Schröder non réduit à une feuille $S = \wedge(S^{(1)}, \dots, S^{(\ell)})$ de manière récursive par

$$\mathcal{S}^{-1} := \begin{cases} 0 & \text{si } S = \wedge(\perp, \perp), \\ \mathcal{S}^{-1}(S^{(1)})' \cdot 0 \cdot \dots \cdot 0 \cdot \mathcal{S}^{-1}(S^{(\ell)})' & \text{sinon,} \end{cases} \quad (6.2.41)$$

où la notation u' signifie que l'on considère le mot u où chacune de ses lettres sont incrémentées.

Il est clair, par la caractérisation des mots de $Schr$ fournie par le lemme 6.2.16, que \mathcal{S} est une bijection de l'ensemble des mots de $Schr$ de longueur n vers l'ensemble des arbres de Schröder à $n+1$ feuilles et que \mathcal{S}^{-1} est son inverse. Un exemple de cette bijection est donné par la figure 6.6.

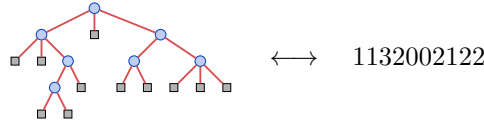


FIGURE 6.6 – Un arbre de Schröder et l'élément correspondant dans l'opérade ns $Schr$.

Cette bijection implique que la série de Hilbert de $Schr$ est la série génératrice des arbres de Schröder (suite A001003 de [Slo]) vérifiant

$$F_{Schr}(t) = \frac{1 - 3t - \sqrt{1 - 6t + t^2}}{4t} = t + 3t^2 + 11t^3 + 45t^4 + 197t^5 + 903t^6 + 4279t^7 + \dots \quad (6.2.42)$$

Notons maintenant respectivement \nearrow , \nearrow et \nwarrow les générateurs 00, 01 et 10 de $Schr$.

Proposition 6.2.18. *Les générateurs $\{\nearrow, \nearrow, \nwarrow\}$ de l'opérade ns $Schr$ sont soumis, en degré deux, uniquement aux relations*

$$\nearrow \circ_1 \nearrow = \nearrow \circ_2 \nearrow, \quad (6.2.43)$$

$$\nearrow \circ_1 \nwarrow = \nwarrow \circ_2 \nearrow, \quad (6.2.44)$$

$$\nwarrow \circ_1 \nearrow = \nearrow \circ_2 \nwarrow, \quad (6.2.45)$$

$$\nwarrow \circ_1 \nwarrow = \nwarrow \circ_2 \nwarrow, \quad (6.2.46)$$

$$\nwarrow \circ_1 \nearrow = \nwarrow \circ_2 \nearrow, \quad (6.2.47)$$

$$\nearrow \circ_1 \nearrow = \nearrow \circ_2 \nearrow, \quad (6.2.48)$$

$$\nwarrow \circ_1 \nwarrow = \nwarrow \circ_2 \nwarrow. \quad (6.2.49)$$

Démonstration. C'est une inspection directe des $3 \times 2 \times 3 = 18$ expressions de degré deux que l'on peut construire sur trois générateurs d'arité deux. \square

Remarquons que la proposition 6.2.18 ne signifie pas que les seules relations non triviales de $Schr$ sont uniquement les sept relations mentionnées.

6.2.8 Une opérade sur les chemins de Motzkin

Soit la sous-opérade ns $Motz$ de TN engendrée par les éléments 00 et 010. Étant donné que les éléments 00 et 01 sont les générateurs de $FCat^{(1)}$ et que $010 = 00 \circ_1 01$, $Motz$ est une sous-opérade ns de $FCat^{(1)}$. Voici les premiers éléments de $Motz$, arité par arité :

- arité 1 : 0 ;

- arité 2 : 00 ;
- arité 3 : 000, 010 ;
- arité 4 : 0000, 0010, 0100, 0110 ;
- arité 5 : 00000, 00010, 00100, 00110, 01000, 01010, 01100, 01110, 01210.

Les éléments de *Motz* sont caractérisés de la manière suivante :

Lemme 6.2.19. *Les éléments de l'opérade ns Motz sont exactement les mots x sur l'alphabet \mathbb{N} qui commencent et se terminent par 0 et tels que $|x_i - x_{i+1}| \leq 1$ pour tout $i \in [|x| - 1]$.*

Démonstration. Montrons d'abord que tout mot de *Motz* vérifie les conditions du lemme. On procède par récurrence sur la longueur des mots. Le cas de base est vérifié car $x := \mathbf{1} = 0$ respecte les conditions du lemme. Soit x un élément de *Motz* d'arité $n \geq 2$. D'après le lemme 6.2.1, x peut s'écrire sous la forme $x = y \circ_i g$ où y est un élément d'arité strictement inférieure à n et $g \in \{00, 010\}$. Nous avons maintenant deux cas à examiner en fonction de g . Si $g = 00$, nous avons

$$x = (y_1, \dots, y_{i-1}, y_i, y_i, y_{i+1}, \dots, y_{|y|}), \quad (6.2.50)$$

et, comme par hypothèse de récurrence y vérifie les conditions du lemme, x les vérifie clairement aussi. Dans l'autre cas, nous avons

$$x = (y_1, \dots, y_{i-1}, y_i, y_i + 1, y_i, y_{i+1}, \dots, y_{|y|}), \quad (6.2.51)$$

et ici encore, le fait que y vérifie par hypothèse de récurrence les conditions du lemme implique que c'est aussi le cas pour x .

Montrons maintenant que *Motz* permet d'engendrer tous les mots qui vérifient les conditions du lemme. Nous procédons par récurrence sur la longueur des mots. Il existe un unique mot $x := 0 = \mathbf{1}$ de longueur 1 qui respecte les conditions du lemme et celui-ci figure bien dans *Motz*. Soit maintenant x un mot de longueur $n \geq 2$ qui satisfait les conditions du lemme. S'il existe dans x un facteur $x_i \cdot x_{i+1}$ avec $x_i = x_{i+1}$, en posant

$$y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n), \quad (6.2.52)$$

nous avons bien $y \circ_i 00 = x$. Ainsi, par hypothèse de récurrence, y est un mot engendré par *Motz*, ce qui implique que x l'est aussi. Sinon, lorsque x n'admet pas de tel facteur, en posant $\mathbf{b} := \max x$, d'après les conditions du lemme que vérifie x , ce dernier admet un facteur $x_i \cdot x_{i+1} \cdot x_{i+2}$ avec $x_{i+1} = \mathbf{b}$ et $x_i = x_{i+2} = \mathbf{b} - 1$. En posant

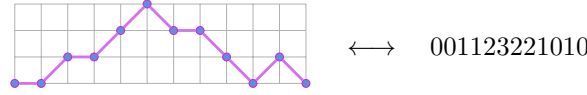
$$y := (x_1, \dots, x_i, x_{i+3}, \dots, x_n), \quad (6.2.53)$$

nous avons $y \circ_i 010 = x$. Par conséquent, comme dans le cas précédent, x est bien engendré par *Motz*. \square

Dans le but de déterminer la série de Hilbert de *Motz*, nous avons besoin de la définition suivante :

Définition 6.2.20. *Un chemin de Motzkin de taille n est un chemin dans le réseau \mathbb{Z}^2 qui relie les points de coordonnées $(0, 0)$ et $(n, 0)$ par des pas montants $(1, 1)$, des pas descendants $(1, -1)$ et des pas stationnaires $(1, 0)$, avec la condition supplémentaire que le chemin reste au dessus de la droite horizontale d'ordonnée 0.*

Les éléments de *Motz* d'arité n sont en bijection avec les chemins de Motzkin de taille $n - 1$ de la manière suivante. À tout élément $x \in \text{Motz}(n)$, on associe une suite $s := s_1 \dots s_{n-1}$ sur l'alphabet $\{-1, 0, 1\}$ telle que $s_i := x_{i+1} - x_i$. Les conditions du lemme 6.2.19 que respecte x impliquent que $s_1 + \dots + s_{n-1} = 0$ et $s_1 + \dots + s_i \geq 0$ pour tout $i \in [n - 1]$. Ainsi, en associant à chaque lettre s_i de s un pas montant si $s_i = 1$, un pas descendant si $s_i = -1$ et un pas

FIGURE 6.7 – Un chemin de Motzkin et l'élément correspondant dans l'opérade ns *Motz*.

stationnaire sinon, un chemin de Motzkin de taille $n - 1$ est obtenu. Cette construction est de plus clairement inversible. Un exemple de cette bijection est donné dans la figure 6.7.

Nous déduisons ainsi de cette bijection que la série de Hilbert de *Motz* est la série génératrice des mots de Motzkin (suite **A001006** de [Slo]) vérifiant

$$F_{\text{Motz}}(t) = \frac{1 - t - \sqrt{-3t^2 - 2t + 1}}{2t} = t + t^2 + 2t^3 + 4t^4 + 9t^5 + 21t^6 + 51t^7 + 127t^8 + \dots \quad (6.2.54)$$

6.2.9 Une opérade sur les compositions d'entiers

Soit la sous-opérade ns *Comp* de TN_2 engendrée par les éléments 00 et 01. Comme $\text{FCat}^{(1)}$ est la sous-opérade ns de TN engendrée par 00 et 01 et que TN_2 est un quotient de TN , *Comp* est un quotient de $\text{FCat}^{(1)}$. Voici les premiers éléments de *Comp*, arité par arité :

- arité 1 : 0 ;
- arité 2 : 00, 01 ;
- arité 3 : 000, 001, 010, 011 ;
- arité 4 : 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111.

Les éléments de *Comp* sont caractérisés de la manière suivante :

Lemme 6.2.21. *Les éléments de l'opérade ns *Comp* sont exactement les mots sur l'alphabet $\{0, 1\}$ dont la première lettre est 0.*

Démonstration. Il est immédiat que tous les mots de *Comp* sont des mots sur l'alphabet $\{0, 1\}$. Il commencent par 0 car les générateurs 00 et 01 commencent par 0.

Montrons que *Comp* permet d'engendrer tous les mots qui vérifient les conditions du lemme. Nous procédons par récurrence sur la longueur des mots. L'unique mot de longueur 1 est $0 = \mathbf{1}$ et il figure bien dans *Comp*. Soit x un mot de longueur $n \geq 2$ qui respecte les conditions du lemme. Si x ne possède aucune occurrence de 1, x est obtenu en réalisant des substitutions partielles du générateur 00 avec lui-même. Sinon, comme la première lettre de x est 0, il existe dans x un facteur $x_i \cdot x_{i+1}$ égal à 01. Nous avons

$$x = (x_1, \dots, x_i = 0, x_{i+1} = 1, \dots, x_n), \quad (6.2.55)$$

et en posant

$$y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n), \quad (6.2.56)$$

nous avons $x = y \circ_i 01$. Le mot y respecte les conditions du lemme, et, par hypothèse de récurrence, il est engendré par *Comp*. Le mot x l'est donc également. \square

Nous déduisons du lemme 6.2.21 que la série de Hilbert de *Comp* est

$$F_{\text{Comp}}(t) = \frac{t}{1 - 2t} = t + 2t^2 + 4t^3 + 8t^4 + 16t^5 + 32t^6 + 64t^7 + \dots \quad (6.2.57)$$

Rappelons qu'une *composition d'entiers* est une suite finie d'entiers $C := (c_1, \dots, c_\ell)$. Les lettres c_i sont appelées *parts* de C et sa *taille* est la somme de ses parts. Il est bien connu que le nombre de compositions de taille $n \geq 1$ est 2^{n-1} .

On peut encoder une composition C par son *diagramme ruban*, où l'on représente chaque part c_i de C par une colonne de c_i cases et où la case la plus en haut de la colonne correspondant à la part c_{i+1} est attachée à droite de la case la plus en bas de la colonne correspondant à la part c_i . On appelle i^e case du diagramme ruban de C la i^e case que l'on rencontre lorsque l'on parcourt les cases du diagramme ruban de C de haut en bas et de gauche à droite. Le *transposé* du diagramme ruban de C est le diagramme ruban dans lequel les colonnes sont changées en lignes et réciproquement.

Il existe une bijection entre les mots de longueur $n \geq 1$ qui satisfont les conditions du lemme 6.2.21 et les compositions d'entiers de taille n . En effet, étant donné un tel mot x , celui-ci se factorise en

$$x = 01^{k_1} \cdot 01^{k_2} \cdot \dots \cdot 01^{k_\ell}, \quad (6.2.58)$$

où $k_i \geq 0$ pour tout $i \in [\ell]$ et on lui associe la composition $(k_1 + 1, k_2 + 1, \dots, k_\ell + 1)$. Ce processus est bien entendu bijectif.

La substitution partielle de $Comp$ peut se reformuler en termes de diagrammes ruban. Soient C et D deux diagrammes ruban de tailles respectives n et m et $i \in [n]$. Nous avons deux cas : si la i^e case de C est la case la plus en haut de sa colonne, le diagramme ruban $C \circ_i D$ est obtenu en remplaçant la i^e case de C par D ; sinon, lorsque il existe une case au dessus de la i^e case de C , le diagramme ruban $C \circ_i D$ est obtenu en remplaçant la i^e case de C par le transposé de D . La figure 6.8 montre deux exemples de substitutions partielles dans $Comp$.

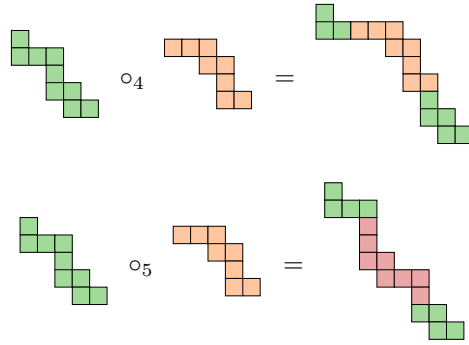


FIGURE 6.8 – Deux exemples de substitutions partielles dans l'opérade ns $Comp$.

Notons à présent respectivement $\begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix}$ et $\begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix}$ les générateurs 00 et 01 de $Comp$.

Proposition 6.2.22. *Les générateurs $\left\{ \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix}, \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \right\}$ de l'opérade ns $Comp$ sont soumis, en degré deux, uniquement aux relations*

$$\begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} \circ_1 \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} = \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} \circ_2 \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix}, \quad (6.2.59)$$

$$\begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \circ_1 \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} = \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} \circ_2 \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix}, \quad (6.2.60)$$

$$\begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \circ_1 \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} = \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \circ_2 \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix}, \quad (6.2.61)$$

$$\begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix} \circ_1 \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} = \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \circ_2 \begin{smallmatrix} \blacksquare \\ \blacksquare \end{smallmatrix}. \quad (6.2.62)$$

Démonstration. C'est une inspection directe des $2 \times 2 \times 2 = 8$ expressions de degré deux que l'on peut construire sur deux générateurs d'arité deux. \square

Remarquons que la proposition 6.2.22 ne signifie pas que les seules relations non triviales de $Comp$ sont uniquement les quatre relations mentionnées.

6.2.10 Une opérade sur les animaux dirigés

Soit la sous-opérade ns AnD de $T\mathbb{N}_3$ engendrée par les éléments 00 et 01. Comme $FCat^{(1)}$ est la sous-opérade ns de $T\mathbb{N}$ engendrée par 00 et 01 et que $T\mathbb{N}_3$ est un quotient de $T\mathbb{N}$, AnD est un quotient de $FCat^{(1)}$. Voici les premiers éléments de AnD , arité par arité :

- arité 1 : 0 ;
- arité 2 : 00, 01 ;
- arité 3 : 000, 001, 010, 011, 012 ;
- arité 4 : 0000, 0001, 0010, 0011, 0012, 0100, 0101, 0110, 0111, 0112, 0120, 0121, 0122 ;
- arité 5 : 00000, 00001, 00010, 00011, 00012, 00100, 00101, 00110, 00111, 00112, 00120, 00121, 00122, 01000, 01001, 01010, 01011, 01012, 01100, 01101, 01110, 01111, 01112, 01120, 01121, 01122, 01200, 01201, 01202, 01210, 01211, 01212, 01220, 01221, 01222.

Avant d'aller plus loin, nous avons besoin de quelques définitions sur les animaux dirigés.

Définition 6.2.23. *Un animal dirigé A est un sous-ensemble de points du réseau \mathbb{N}^2 qui contient le point $(0,0)$ et tel que*

$$(i, j) \in A \text{ avec } i \geq 1 \text{ ou } j \geq 1 \quad \text{implique} \quad (i-1, j) \in A \text{ ou } (i, j-1) \in A. \quad (6.2.63)$$

La taille d'un animal dirigé est le nombre de points qui le constituent

Les animaux dirigés sont habituellement représentés sous forme de diagrammes. Chaque point (i, j) d'un animal dirigé est encodé par un point d'abscisse i et d'ordonnée j . Le point $(0,0)$ est dessiné le plus en bas et à gauche. La figure 6.9 montre un animal dirigé.

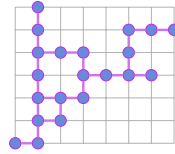


FIGURE 6.9 – Un animal dirigé de taille 21.

Soit \mathcal{L}_n l'ensemble des mots x de longueur n sur l'alphabet $\{-1, 0, 1\}$ tels que, pour tout préfixe $x_1 \dots x_\ell$ de x , l'entier $x_1 + \dots + x_\ell$ est positif.

Une bijection entre les animaux dirigés de taille n et les éléments de \mathcal{L}_{n-1} fut construite par Gouyou-Beauchamps et Viennot en [GBV88]. Il ne nous est pas nécessaire ici de la détailler, nous nous appuyons dans ce qui suit uniquement sur le fait que ces deux ensembles sont en bijection.

Introduisons l'application

$$\phi : \{0, 1, 2\}^2 \rightarrow \{-1, 0, 1\}, \quad (6.2.64)$$

définie par

$$\phi(\mathbf{a} \cdot \mathbf{b}) := \begin{cases} \mathbf{b} - \mathbf{a} & \text{si } |\mathbf{b} - \mathbf{a}| \leq 1, \\ 1 & \text{si } \mathbf{a} = 2 \text{ et } \mathbf{b} = 0, \\ -1 & \text{sinon } (\mathbf{a} = 0 \text{ et } \mathbf{b} = 2). \end{cases} \quad (6.2.65)$$

Cette application peut se comprendre de la manière suivante. Étant donné que -1 est congru à 2 modulo 3, nous pouvons représenter l'élément 2 de \mathbb{N}_3 par l'élément -1 . Avec cet encodage, ϕ s'exprime simplement par $\phi(\mathbf{a} \cdot \mathbf{b}) = \mathbf{b} - \mathbf{a}$.

Nous étendons ϕ en l'application

$$\phi : \{0, 1, 2\}^n \rightarrow \{-1, 0, 1\}^{n-1}, \quad (6.2.66)$$

où l'on pose, pour tous $\mathbf{a}, \mathbf{b} \in \{0, 1, 2\}$ et $u \in \{0, 1, 2\}^*$, $\phi(\mathbf{a}) := \epsilon$ et

$$\phi(\mathbf{a} \cdot \mathbf{b} \cdot u) := \phi(\mathbf{a} \cdot \mathbf{b}) \cdot \phi(\mathbf{b} \cdot u). \quad (6.2.67)$$

Lemme 6.2.24. *Si x est un élément de $AnD(n)$, alors le mot $\phi(x)$ appartient à \mathcal{L}_{n-1} .*

Démonstration. On procède par récurrence sur la longueur de x . Si $|x| = 1$, alors $x = \mathbf{1} = 0$, et comme l'élément $\phi(0) = \epsilon$ figure dans \mathcal{L}_0 , le lemme est vérifié.

Sinon, lorsque x est de longueur $n \geq 2$, d'après le lemme 6.2.1, il existe $y \in AnD(n-1)$, $g \in \{00, 01\}$ et $i \in [n]$ tels que $x = y \circ_i g$. Si $g = 00$, nous avons

$$x = (y_1, \dots, y_i, y_i, \dots, y_{n-1}), \quad (6.2.68)$$

et, d'après la définition de ϕ et l'hypothèse de récurrence, le mot $\phi(x)$ est obtenu à partir de $\phi(y) \in \mathcal{L}_{n-2}$ en y insérant à un endroit approprié la lettre 0. Nous avons donc $\phi(x) \in \mathcal{L}_{n-1}$. Si $g = 01$, nous avons en revanche

$$x = (y_1, \dots, y_i, y_i + 1, \dots, y_{n-1}), \quad (6.2.69)$$

et deux sous-cas à explorer. Si $y_i + 1$ est la dernière lettre de x , alors, d'après la définition de ϕ et l'hypothèse de récurrence, le mot $\phi(x)$ est obtenu à partir de $\phi(y) \in \mathcal{L}_{n-2}$ en y concaténant à sa droite la lettre 1. Nous avons donc $\phi(x) \in \mathcal{L}_{n-1}$. Dans le cas contraire, d'après la définition de ϕ , le mot $\phi(x)$ est obtenu à partir de $\phi(y)$ en substituant la lettre qui correspond à $\phi(y_i \cdot y_{i+1})$ par le facteur $\phi(y_i \cdot y_i + 1) \cdot \phi(y_i + 1 \cdot y_{i+1})$. Encore d'après la définition de ϕ et un simple calcul, nous obtenons l'inégalité

$$\phi(y_i \cdot y_{i+1}) \leq \phi(y_i \cdot y_i + 1) + \phi(y_i + 1 \cdot y_{i+1}), \quad (6.2.70)$$

pour toutes les valeurs possibles de y_i et de y_{i+1} . Finalement, comme $\phi(y_i \cdot y_i + 1) = -1$ implique $\phi(y_i \cdot y_{i+1}) = -1$, et d'après l'hypothèse de récurrence, nous avons bien $\phi(x) \in \mathcal{L}_{n-1}$. \square

Proposition 6.2.25. *L'application ϕ définit une bijection entre $AnD(n)$ et \mathcal{L}_{n-1} .*

Démonstration. Le lemme 6.2.24 montre que ϕ est bien définie, i.e., il s'agit bien d'une application de $AnD(n)$ dans \mathcal{L}_{n-1} .

L'injectivité de ϕ provient du fait que tout mot de AnD commence par 0 (car ses générateurs commencent par 0) et du fait que si $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \{0, 1, 2\}$ sont telles que $\mathbf{b} \neq \mathbf{c}$, alors $\phi(\mathbf{a} \cdot \mathbf{b}) \neq \phi(\mathbf{a} \cdot \mathbf{c})$.

Montrons à présent que ϕ est surjective. Nous procédons par récurrence sur la longueur des mots pour montrer que tout élément de \mathcal{L}_{n-1} admet un antécédent dans $AnD(n)$. Le cas de base est vérifié car $\mathcal{L}_0 = \{\epsilon\}$, $\phi(0) = \epsilon$ et $0 = \mathbf{1} \in AnD$. Soit maintenant u un élément de \mathcal{L}_{n-1} . Il est évident, d'après la définition de ϕ , qu'il existe un élément de $TN_3(n)$ tel que $\phi(x) = u$. Montrons que x est également un élément de AnD . Nous trois cas à examiner suivant la dernière lettre \mathbf{a} de u .

Cas 1. Si $\mathbf{a} = 0$, alors x se termine par un suffixe 00, 11, ou 22. Posons $y := (x_1, \dots, x_{n-1})$. Nous avons $x = y \circ_{n-1} 00$, et, comme $\phi(y)$ est un préfixe de $\phi(x)$, y est par hypothèse de récurrence un élément de $AnD(n-1)$. Ceci implique donc que x est un élément de AnD .

Cas 2. Si $\mathbf{a} = 1$, alors x se termine par un suffixe 01, 12, ou 20. Posons $y := (x_1, \dots, x_{n-1})$. Nous avons $x = y \circ_{n-1} 01$, et, comme $\phi(y)$ est ici aussi un préfixe de $\phi(x)$, y est par hypothèse de récurrence un élément de $AnD(n-1)$. Ceci implique donc que x est un élément de AnD .

Cas 3. Si $\mathbf{a} = -1$, il existe dans u une occurrence de 1. Ceci implique l'existence d'un facteur $x_i \cdot x_{i+1}$ égal à 01, 12, ou 20 dans x . Prenons i maximal. Comme la dernière lettre de u est -1 , il existe une lettre à droite de x_{i+1} . La maximalité de i implique qu'il y a exactement six possibilités pour le facteur $x_i \cdot x_{i+1} \cdot x_{i+2}$ qui se regroupent en deux cas.

Cas 3.1. Si $x_i \cdot x_{i+1} \cdot x_{i+2} \in \{011, 122, 200\}$, on pose $y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n)$. Comme dans ce cas, $\phi(x_i \cdot x_{i+2}) = \phi(x_i \cdot x_{i+1}) + \phi(x_{i+1} \cdot x_{i+2})$, on a $y \in \mathcal{L}_{n-2}$, et par hypothèse de récurrence, y est dans AnD . Étant donné que $x = y \circ_{i+1} 00$, x l'est également.

Cas 3.2. Si $x_i \cdot x_{i+1} \cdot x_{i+2} \in \{010, 121, 202\}$, on pose $y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n)$. Comme dans ce cas, $\phi(x_i \cdot x_{i+2}) = \phi(x_i \cdot x_{i+1}) + \phi(x_{i+1} \cdot x_{i+2})$, on a $y \in \mathcal{L}_{n-2}$, et par hypothèse de récurrence, y est dans AnD . Étant donné que $x = y \circ_i 01$, x l'est également. \square

La proposition 6.2.25 montre que l'on peut interpréter les éléments de AnD d'arité n comme des animaux dirigés de taille n . La série de Hilbert de AnD est donc de la forme

$$F_{AnD}(t) = \frac{2t}{3t - 1 + \sqrt{1 - 2t - 3t^2}} - 1 = t + 2t^2 + 5t^3 + 13t^4 + 35t^5 + 96t^6 + 267t^7 + \dots, \quad (6.2.71)$$

et ses coefficients forment la suite [A005773](#) de [Slo].

6.2.11 Une opérade sur les compositions d'entiers segmentées

Soit la sous-opérade ns $SComp$ de TN_3 engendrée par les éléments 00, 01 et 02. Comme $FCat^{(2)}$ est la sous-opérade ns de TN engendrée par 00, 01 et 02 et que TN_3 est un quotient de TN , $SComp$ est un quotient de $FCat^{(2)}$. Voici les premiers éléments de $SComp$, arité par arité :

- arité 1 : 0 ;
- arité 2 : 00, 01, 02 ;
- arité 3 : 000, 001, 002, 010, 011, 012, 020, 021, 022.

Les éléments de $SComp$ sont caractérisés de la manière suivante :

Lemme 6.2.26. *Les éléments de l'opérade ns $SComp$ sont exactement les mots sur l'alphabet $\{0, 1, 2\}$ dont la première lettre est 0.*

Démonstration. Il est immédiat que tous les mots de $SComp$ sont des mots sur l'alphabet $\{0, 1, 2\}$. Il commencent par 0 car les générateurs 00, 01 et 02 commencent par 0.

Montrons que $SComp$ permet d'engendrer tous les mots qui vérifient les conditions du lemme. Nous procédons par récurrence sur la longueur des mots. L'unique mot de longueur 1 est $0 = 1$ et il figure bien dans $SComp$. Soit x un mot de longueur $n \geq 2$ qui respecte les conditions du lemme. Si x ne possède aucune occurrence ni de 1 ni de 2, x est obtenu en réalisant des substitutions partielles du générateur 00 avec lui-même. Sinon, comme la première lettre de x est 0, il existe dans x un facteur $x_i \cdot x_{i+1} =: g$ égal à 01 ou à 02. En posant

$$y := (x_1, \dots, x_i, x_{i+2}, \dots, x_n), \quad (6.2.72)$$

nous avons $x = y \circ_i g$. Le mot y respecte les conditions du lemme, et, par hypothèse de récurrence, il est engendré par $SComp$. Le mot x l'est donc également. \square

Nous déduisons du lemme 6.2.26 que la série de Hilbert de $SComp$ est

$$F_{SComp}(t) = \frac{t}{1 - 3t} = t + 3t^2 + 9t^3 + 27t^4 + 81t^5 + 243t^6 + \dots \quad (6.2.73)$$

Une *composition d'entiers segmentée* [NT07] est une suite finie $S := (S_1, \dots, S_\ell)$ de compositions d'entiers. Sa taille est la somme des tailles des compositions d'entiers S_i pour tout $i \in [n]$. Le nombre de compositions d'entiers segmentées de taille $n \geq 1$ est 3^{n-1} .

Une composition d'entiers segmentée $S := (S_1, \dots, S_\ell)$ peut être encodée par un *diagramme ruban*. On considère pour cela les diagrammes ruban des compositions d'entiers S_1, \dots, S_ℓ où l'on marque les premières cases de chacun d'eux. On colle ensuite ensemble ces diagrammes de

gauche à droite, *i.e.*, pour tout $i \in [\ell - 1]$, on juxtapose la première case du diagramme de S_{i+1} juste à droite de la dernière case du diagramme de S_i .

Il existe une bijection simple entre les mots de longueur $n \geq 1$ qui satisfont les conditions du lemme 6.2.26 et les compositions d'entiers segmentées de taille n . Étant donné un tel mot x , celui-ci se factorise en

$$x = 0u^{(1)} \cdot 0u^{(2)} \cdot \dots \cdot 0u^{(\ell)}, \quad (6.2.74)$$

où les mots $u^{(i)}$ sont sur l'alphabet $\{1, 2\}$. On associe à x la composition d'entiers segmentée (S_1, \dots, S_ℓ) où pour tout $i \in [\ell]$, S_i est la composition d'entiers associée au mot $0u^{(i)'}$ selon (6.2.58), où $u^{(i)'}$ est le mot $u^{(i)}$ dans lequel ses lettres sont décrémentées.

De la même manière que ce que nous avons fait dans le cas de l'opérade ns *Comp*, la substitution partielle de *SComp* peut être interprétée en termes de diagrammes de compositions d'entiers segmentées.

6.2.12 L'opérade diassociative

Soit $M := \{0, 1\}$ le sous-monoïde du monoïde multiplicatif. Soit \mathcal{P} la sous-opérade ns de *TM* engendrée par l'ensemble $\{01, 10\}$. Voici les premiers éléments de \mathcal{P} , arité par arité :

- arité 1 : 1 ;
- arité 2 : 01, 10 ;
- arité 3 : 001, 010, 100 ;
- arité 4 : 0001, 0010, 0100, 1000.

L'objectif de ce paragraphe est de montrer que \mathcal{P} est isomorphe à l'opérade *diassociative*. Rappelons que cette opérade, introduite par Loday [Lod01] (voir aussi [Zin10]), est engendrée par deux éléments d'arité deux \dashv et \vdash , appelés respectivement *gauche* et *droite*, et soumis aux relations suivantes :

$$\dashv \circ_1 \dashv = \dashv \circ_2 \dashv = \dashv \circ_2 \vdash, \quad (6.2.75)$$

$$\dashv \circ_1 \vdash = \vdash \circ_2 \dashv, \quad (6.2.76)$$

$$\vdash \circ_2 \vdash = \vdash \circ_1 \vdash = \vdash \circ_1 \dashv. \quad (6.2.77)$$

Cette opérade — que l'on note *Dias* — est non symétrique, ensembliste, binaire et quadratique et est le dual de Koszul de l'opérade ns dendriforme (voir le paragraphe 3.2.4 du chapitre 3). La série de Hilbert de *Dias* est

$$F_{Dias}(t) = \frac{t}{(1-t)^2} = t + 2t^2 + 3t^3 + 4t^4 + \dots \quad (6.2.78)$$

Les éléments de \mathcal{P} sont caractérisés de la manière suivante :

Lemme 6.2.27. *Les éléments de l'opérade ns \mathcal{P} sont exactement les mots sur l'alphabet $\{0, 1\}$ qui possèdent une unique lettre 1.*

Démonstration. On montre facilement par récurrence sur la longueur des mots, et en utilisant le lemme 6.2.1, que tout élément de \mathcal{P} respecte la condition du lemme.

Montrons par récurrence sur la longueur des mots que \mathcal{P} contient tous les mots qui vérifient la condition du lemme. Soit x un mot sur l'alphabet $\{0, 1\}$ possédant une unique occurrence de 1. Si $x = 1$, x est bien un élément de \mathcal{P} puisque $x = \mathbf{1}$. Sinon, il existe dans x un facteur $x_i \cdot x_{i+1}$ de x égal à 01 ou 10. Sans perte de généralité, supposons $x_i \cdot x_{i+1} = 01$ et posons

$$y := (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_{|x|}). \quad (6.2.79)$$

Le mot y possède clairement une unique occurrence de 1, et ainsi, par hypothèse de récurrence, y est un élément de \mathcal{P} . Étant donné que $x = y \circ_i 01$, x est également un élément de \mathcal{P} . \square

Nous déduisons du lemme 6.2.27 que les opérades \mathcal{P} et $Dias$ ont la même série de Hilbert.

Proposition 6.2.28. *Les opérades ns $Dias$ et \mathcal{P} sont isomorphes.*

Démonstration. Considérons l'opérade ns ensembliste $\mathcal{Q} := \bigsqcup_{n \geq 1} \mathcal{Q}(n)$ où

$$\mathcal{Q}(n) := \{(j, n) : j \in [n]\}, \quad (6.2.80)$$

et la substitution partielle est définie par

$$(j, n) \circ_i (k, m) := \begin{cases} (j + m - 1, n + m - 1) & \text{si } i < j, \\ (j + k - 1, n + m - 1) & \text{si } i = j, \\ (j, n + m - 1) & \text{si } i > j. \end{cases} \quad (6.2.81)$$

Il est connu (voir [Cha05]) que l'opérade $Dias$ est isomorphe à \mathcal{Q} pour l'isomorphisme d'opérade $\phi : Dias \rightarrow \mathcal{Q}$ défini par $\phi(-) := (1, 2)$ et $\phi(\vdash) := (2, 2)$.

Considérons à présent l'application $\psi : \mathcal{P} \rightarrow \mathcal{Q}$ définie par tout $u := 0 \dots 010 \dots 0 \in \mathcal{P}(n)$ par $\psi(0 \dots 1 \dots 0) := (j, n)$ où la lettre 1 est en j^{e} position dans u . D'après le lemme 6.2.27, ψ est bien définie et est de plus une bijection. Une inspection directe montre que ψ est un morphisme d'opérade. Ainsi, \mathcal{P} et $Dias$ sont isomorphes pour l'isomorphisme d'opérade $\phi^{-1} \circ \psi : \mathcal{P} \rightarrow Dias$ qui vérifie $(\phi^{-1} \circ \psi)(10) = \dashv$ et $(\phi^{-1} \circ \psi)(01) = \vdash$. \square

6.3 L'algèbre de Hopf de l'opérade des arbres plans enracinés

Nous appliquons ici, à l'opérade APE des arbres plans enracinés définie dans le paragraphe 6.2.5, la construction rappelée dans le paragraphe 3.3.2 du chapitre 3 qui sur la donnée d'une opérade ensembliste produit une algèbre de Hopf non commutative.

6.3.1 L'algèbre de Hopf des forêts planes enracinées

Forêts ordonnées d'arbres plans enracinés

Commençons par quelques définitions sur les forêts ordonnées d'arbres plans enracinés. Une *forêt ordonnée d'arbres plans enracinés* F est une suite d'arbres plans enracinés qui possèdent chacun au moins deux nœuds. La *taille* de F est son nombre d'arêtes. Dans ce qui suit, nous appelons simplement *forêt* une forêt ordonnée d'arbres plans enracinés et *arbre* un arbre plan enraciné.

On appelle i^{e} *nœud* d'une forêt F le i^{e} nœud que l'on rencontre en visitant les nœuds de F selon un parcours en profondeur gauche droite. Par convention, la racine de l'arbre le plus à gauche de F est son 0^{e} nœud. Pour faire référence aux arêtes de F , on note par r_{ij} toute arête de F qui connecte son i^{e} nœud à son j^{e} nœud. Par convention, on impose $i > j$. Avec cette notation, nous pouvons encoder une forêt par l'ensemble des éléments r_{ij} qui sont des arêtes de F . Par extension, on note $r_{ij} \in F$ si r_{ij} est une arête de F . La figure 6.10 illustre un exemple d'un tel encodage.

On appelle i^{e} *arête* de F la i^{e} arête que l'on rencontre en visitant les arêtes de F selon un parcours en profondeur. Par convention, la première arête visitée est la 1^{e} arête de F .

Une arête r_{ij} de F est dite *attachée* au j^{e} nœud de F . L'*extrémité* de r_{ij} est le i^{e} nœud de F . L'arête r_{ij} est l'*arête mère* d'une arête $r_{k\ell}$ de F si $\ell = i$. Autrement dit, l'arête $r_{k\ell}$ est attachée à l'extrémité de r_{ij} dans F . L'arête r_{ij} est l'*arête sœur gauche* de $r_{k\ell}$ si $j = \ell$ et $i < k$. Autrement dit, les arêtes r_{ij} et $r_{k\ell}$ sont toutes deux attachées au j^{e} nœud de F et le i^{e} nœud est à gauche du k^{e} nœud dans F . La notion d'*arête sœur droite* est définie de manière symétrique.

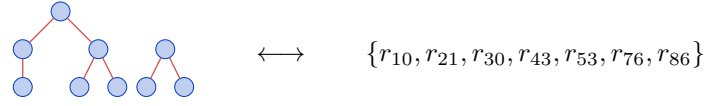


FIGURE 6.10 – Une forêt ordonnée d'arbres plans enracinés et son ensemble d'arêtes.

Éléments de base

Considérons l'algèbre de Hopf $\mathbf{APE} := \mathfrak{H}_{APE}$. Voici ses premiers éléments de base :

- degré 0 : \mathbf{E}_0 ;
- degré 1 : \mathbf{E}_{01} ;
- degré 2 : \mathbf{E}_{011} , \mathbf{E}_{012} , $\mathbf{E}_{01|01}$;
- degré 3 : \mathbf{E}_{0111} , \mathbf{E}_{0112} , \mathbf{E}_{0121} , \mathbf{E}_{0122} , \mathbf{E}_{0123} , $\mathbf{E}_{011|01}$, $\mathbf{E}_{012|01}$, $\mathbf{E}_{01|011}$, $\mathbf{E}_{01|012}$, $\mathbf{E}_{01|01|01}$.

D'après la construction du paragraphe 3.3.2 du chapitre 3, et d'après l'interprétation des éléments de l'opérade APE en termes d'arbres plans enracinés évoquée dans le paragraphe 6.2.5, nous pouvons interpréter les éléments de la base élémentaire de \mathbf{APE} en termes de forêts. Avec cette interprétation, le degré d'un élément \mathbf{E}_F de \mathbf{APE} est donné par le nombre d'arêtes de la forêt F et les éléments indécomposables de \mathbf{APE} sont les \mathbf{E}_T où T est un arbre.

Série de Hilbert

D'après le lemme 3.3.2, et étant donné l'expression la série de Hilbert de APE donnée en (6.2.22) dans le chapitre 6, celle de \mathbf{APE} vérifie

$$F_{\mathbf{APE}}(t) = \frac{2t}{4t - 1 + \sqrt{1 - 4t}} \quad (6.3.1)$$

$$= 1 + \sum_{n \geq 1} \binom{2n-1}{n} t^n \quad (6.3.2)$$

$$= 1 + t + 3t^2 + 10t^3 + 35t^4 + 126t^5 + 462t^6 + 1716t^7 + 6435t^8 + \dots \quad (6.3.3)$$

Exemples de produits et de coproduits

Voyons à présent quelques exemples. Nous avons dans \mathbf{APE}

$$\mathbf{E}_{0122|01} \cdot \mathbf{E}_{011|01} = \mathbf{E}_{0122|01|011|01}, \quad (6.3.4)$$

et, lorsque l'on interprète les éléments de (6.3.4) en termes de forêts, nous obtenons :

$$\mathbf{E}_{\text{forest1}} \cdot \mathbf{E}_{\text{forest2}} = \mathbf{E}_{\text{forest3}}. \quad (6.3.5)$$

The diagram shows the product of two forests. The first forest has two trees: one with two nodes and one with one node. The second forest has two trees: one with two nodes and one with one node. The product is a forest with four trees: one with two nodes, one with two nodes, one with one node, and one with one node.

Nous avons également

$$\begin{aligned} \Delta(\mathbf{E}_{012122}) &= 1 \otimes \mathbf{E}_{012122} + \mathbf{E}_{01} \otimes \mathbf{E}_{012|011} + \mathbf{E}_{011} \otimes \mathbf{E}_{01|011} + \mathbf{E}_{012} \otimes \mathbf{E}_{012|01} \\ &\quad + \mathbf{E}_{0112} \otimes \mathbf{E}_{01|01} + \mathbf{E}_{0121} \otimes \mathbf{E}_{011} + \mathbf{E}_{0122} \otimes \mathbf{E}_{012} + \mathbf{E}_{01122} \otimes \mathbf{E}_{01} \\ &\quad + \mathbf{E}_{01212} \otimes \mathbf{E}_{01} + \mathbf{E}_{012122} \otimes 1, \end{aligned} \quad (6.3.6)$$

et, de même, lorsque l'on interprète les éléments de (6.3.6) en termes de forêts, nous obtenons :

$$\begin{aligned} \Delta \left(\begin{array}{c} \text{E} \\ \text{arbre} \end{array} \right) = & 1 \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \\ & + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \\ & + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes \begin{array}{c} \text{E} \\ \text{arbre} \end{array} + \begin{array}{c} \text{E} \\ \text{arbre} \end{array} \otimes 1. \end{aligned} \quad (6.3.7)$$

Une définition alternative du coproduit

Nous définissons à présent sur l'ensemble $R := \{r_{ij} : 0 \leq j < i\}$ de toutes les arêtes possibles la relation binaire \rhd par

$$r_{ij} \rhd r_{k\ell} \quad \text{si et seulement si} \quad j = k \quad \text{ou} \quad j = \ell \text{ et } i < k. \quad (6.3.8)$$

Un idéal d'une forêt F pour la relation \rhd est un ensemble I d'arêtes de F tel que

$$r_{ij} \in I, r_{k\ell} \in F, \text{ et } r_{ij} \rhd r_{k\ell} \quad \text{impliquent} \quad r_{k\ell} \in I. \quad (6.3.9)$$

Lemme 6.3.1. *Soit T un arbre et I un idéal de T . Alors, les arêtes de I forment un arbre dont la racine est celle de T . De plus, si une arête de T appartient à I alors, si elle existe, sa sœur droite est également dans I .*

Démonstration. Montrons la première partie du lemme. Si une arête r_{ij} appartient à I et qu'elle possède une arête mère $r_{k\ell}$, alors, comme ces deux arêtes possèdent un nœud en commun et d'après notre notation des arêtes, nous avons $j = k$. Par conséquent $r_{ij} \rhd r_{k\ell}$, ce qui implique $r_{k\ell} \in I$. Ceci montre que si une arête est dans I alors toutes ses arêtes ancêtres le sont également. Le sous-graphe de T induit par I est donc un arbre, de racine égale à celle de T .

Montrons la seconde partie du lemme. Si une arête r_{ij} appartient à I et qu'elle possède une arête sœur droite $r_{k\ell}$ alors, comme ces deux arêtes sont sœurs, elles partagent le même nœud parent et donc $j = \ell$. De plus, comme les arêtes sont numérotées selon le parcours en profondeur gauche droite de T , on a $i < k$ et par conséquent $r_{ij} \rhd r_{k\ell}$, ce qui montre $r_{k\ell} \in I$. \square

On appelle *coloriage admissible* d'un arbre T tout idéal de T . Un coloriage admissible I permet, en s'appuyant sur le lemme 6.3.1, de construire à partir de T un couple $T|_I := (S, F)$ où S est l'arbre T restreint aux arêtes de I et à leurs nœuds adjacents et où F est la forêt composée des arêtes de T qui ne sont pas dans I ainsi que de leurs nœuds adjacents. Chaque composante connexe de F forme un arbre dont la racine est son unique nœud qui connectait dans T la racine de T par le chemin le plus court. Ces arbres apparaissent dans F de gauche à droite dans le même ordre qu'il apparaissent en tant que sous-arbres dans T . La figure 6.11 montre un exemple de coloriage admissible et le couple que l'on obtient à partir de celui-ci.

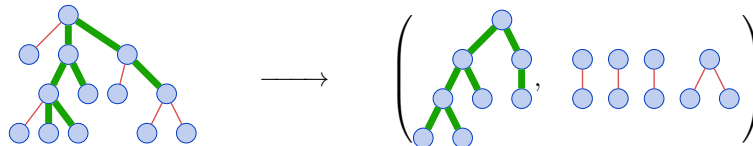


FIGURE 6.11 – Un arbre plan enraciné T et le couple $T|_I := (S, F)$ où I est le coloriage admissible qui contient les arêtes r_{20} , r_{32} , r_{53} , r_{63} , r_{72} , r_{80} et $r_{10,8}$.

Proposition 6.3.2. *Soit T un arbre. Un tenseur de la forme $\mathbf{E}_S \otimes \mathbf{E}_F$ apparaît dans $\Delta(\mathbf{E}_T)$ avec un coefficient à $1_{\mathbb{K}}$ si et seulement s'il existe un coloriage I de T tel que $T|_I = (S, F)$.*

Démonstration. Soient S un arbre et F une forêt constituée de gauche à droite des arbres F_1, \dots, F_ℓ . Par définition du coproduit de **APE**, un tenseur $\mathbf{E}_S \otimes \mathbf{E}_F$ apparaît dans $\Delta(\mathbf{E}_T)$ si et seulement si nous avons l'écriture

$$T = S \circ [G_1, \dots, G_k], \quad (6.3.10)$$

où G_1, \dots, G_k est une suite d'arbres telle qu'il existe un ensemble d'indices

$$E := \{i_1 < \dots < i_\ell\} \quad (6.3.11)$$

vérifiant $G_{i_1} = F_1, \dots, G_{i_\ell} = F_\ell$ et G_j est la forêt vide pour tout $j \notin E$. Ceci est équivalent à dire, d'après l'expression de la substitution partielle en termes d'arbres fournie par le lemme 6.2.11 et d'après le lemme 6.3.1, qu'il existe un unique coloriage I de T tel que $T|_I = (S, F)$. Ceci montre la proposition. \square

Notons $\text{Adm}(T)$ l'ensemble des coloriages admissibles de l'arbre T . Nous pouvons tirer de la proposition 6.3.2 la reformulation du coproduit de **APE** suivante, similaire à celle du coproduit de l'algèbre de Hopf de Connes-Kreimer (voir le paragraphe 2.2.2 du chapitre 2) :

$$\Delta(\mathbf{E}_T) = \sum_{\substack{I \in \text{Adm}(T) \\ (S, F) := T|_I}} \mathbf{E}_S \otimes \mathbf{E}_F. \quad (6.3.12)$$

6.3.2 Une réalisation polynomiale

La F -compatibilité

Posons maintenant $A := \{a_{ij} : 0 \leq j < i\}$ un alphabet infini de variables bi-indexées et non commutatives. Nous définissons de plus sur A la relation binaire \prec par

$$a_{ij} \prec a_{kl} \quad \text{si et seulement si} \quad r_{ij} \rhd r_{kl}. \quad (6.3.13)$$

Muni de cette relation, A est un \prec -alphabet au sens de la définition donnée dans le paragraphe 2.1.4 du chapitre 2.

Soit F une forêt à n arêtes. Un mot $u := u_1 \dots u_n$ sur A de longueur n est F -compatible si, pour tous $i, j \in [n]$, en notant r (resp. r') la i^e (resp. j^e) arête de F ,

$$r \rhd r' \quad \text{implique} \quad u_i \prec u_j. \quad (6.3.14)$$

On note dans ce cas $u \Vdash F$.

Considérons par exemple la forêt

$$F := \begin{array}{c} \bullet \\ | \\ \bullet \\ / \quad \backslash \\ \bullet \quad \bullet \end{array} \quad \begin{array}{c} \bullet \\ | \\ \bullet \\ / \quad \backslash \\ \bullet \quad \bullet \end{array} . \quad (6.3.15)$$

Les mots F -compatibles $u := u_1 u_2 u_3 u_4 u_5$ doivent nécessairement vérifier $u_2 \prec u_1$, $u_2 \prec u_3$, $u_3 \prec u_1$ et $u_4 \prec u_5$. Ainsi, entre autres, les mots $a_{10} a_{21} a_{31} a_{54} a_{64}$, $a_{10} a_{21} a_{31} a_{10} a_{20}$, et $a_{30} a_{10} a_{20} a_{72} a_{82}$ sont F -compatibles.

Un morphisme vers l'algèbre associative libre

Soit $r_A : \mathbf{APE} \rightarrow \mathbb{K}\langle A \rangle$ le morphisme d'algèbre défini linéairement pour tout arbre T par

$$r_A(\mathbf{E}_T) := \sum_{\substack{u \in A^* \\ u \Vdash T}} u. \quad (6.3.16)$$

Notons que (6.3.16) définit r_A sur \mathbf{APE} tout entier car cette dernière est engendrée librement par les éléments \mathbf{E}_T où T est un arbre. Voici quelques exemples :

$$r_A(\mathbf{E}_{01}) = r_A\left(\mathbf{E}_{\begin{smallmatrix} \bullet \\ \bullet \end{smallmatrix}}\right) = \sum_{u_1} u_1 = \sum_{0 \leq j < i} a_{ij}, \quad (6.3.17)$$

$$r_A(\mathbf{E}_{011}) = r_A\left(\mathbf{E}_{\begin{smallmatrix} \bullet \\ \bullet \bullet \end{smallmatrix}}\right) = \sum_{u_1 \prec u_2} u_1 u_2 = \sum_{0 \leq k < j < i} a_{ij} a_{jk} + \sum_{0 \leq j < i < k} a_{ij} a_{kj}, \quad (6.3.18)$$

$$r_A(\mathbf{E}_{012}) = r_A\left(\mathbf{E}_{\begin{smallmatrix} \bullet \\ \bullet \bullet \end{smallmatrix}}\right) = \sum_{u_2 \prec u_1} u_1 u_2 = \sum_{0 \leq j < i < k} a_{ji} a_{kj} + \sum_{0 \leq j < k < i} a_{ij} a_{kj}, \quad (6.3.19)$$

$$r_A(\mathbf{E}_{01|01}) = r_A\left(\mathbf{E}_{\begin{smallmatrix} \bullet \bullet \\ \bullet \bullet \end{smallmatrix}}\right) = \left(\sum_{u_1} u_1\right) \left(\sum_{u_1} u_1\right) = \sum_{\substack{0 \leq j < i \\ 0 \leq k < \ell}} a_{ij} a_{k\ell}. \quad (6.3.20)$$

Un algorithme d'étiquetage

Dans l'objectif de montrer que le morphisme r_A est une réalisation polynomiale de \mathbf{APE} , nous avons besoin de l'ingrédient suivant. Nous introduisons dans ce paragraphe un algorithme qui, sur l'entrée d'un arbre à n nœuds, retourne un arbre de même forme et étiqueté sur l'ensemble $\{0, \dots, n-1\}$.

Algorithme : ÉTIQUETAGEMIN.

Entrée : un arbre plan enraciné T non étiqueté.

Sortie : une version de T étiquetée sur \mathbb{N} .

1. étiqueter la racine de T par 0.
2. Tant qu'il reste au moins un nœud non étiqueté dans T :
 - (a) pour tout nœud x non étiqueté dans T :
 - i. si x est le frère droit d'un nœud étiqueté par i , étiqueter x par $i+1$.
 - ii. Sinon, si x est le fils le plus à gauche d'un nœud étiqueté par i , étiqueter x par $i+1$.

Fin.

La figure 6.12 montre une exécution de l'algorithme ÉTIQUETAGEMIN.

Nous associons à tout arbre T à $n+1$ nœuds le mot sur l'alphabet A

$$m(T) := u_1 \dots u_n, \quad (6.3.21)$$

où $u_i := a_{k\ell}$ si la i^{e} arête de la version de T étiquetée par l'algorithme ÉTIQUETAGEMIN connecte deux nœuds dont le père est étiqueté par ℓ et le fils par k . Par exemple, pour l'arbre considéré dans la figure 6.12, nous obtenons le mot $a_{10} a_{21} a_{31} a_{43} a_{20} a_{32} a_{42} a_{54} a_{52}$. De plus, si F est une forêt, nous lui associons de la même façon le mot

$$m(F) := m(T_1) \cdot \dots \cdot m(T_r), \quad (6.3.22)$$

où F est la concaténation des arbres T_1, \dots, T_r .

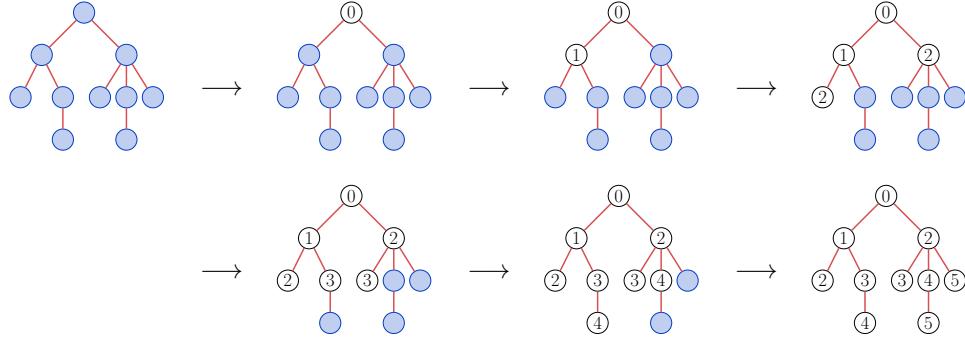


FIGURE 6.12 – Un exemple d'exécution de l'algorithme ÉTIQUETAGEMIN.

Démonstration de la réalisation polynomiale

Lemme 6.3.3. *Soit F une forêt. Alors, le mot $m(F)$ est un monôme de $r_A(\mathbf{E}_F)$.*

Démonstration. Considérons un nœud de F et ses fils après exécution de l'algorithme ÉTIQUETAGEMIN. Ils sont étiquetés de la manière suivante



où k est le nombre de fils du nœud étiqueté par i . Soit maintenant r (resp. r') la j^e (resp. j'^e) arête de F . Si r' est la sœur droite de r , alors, d'après (6.3.23), la j^e lettre de $m(F) =: u$ est de la forme a_{xy} et la j'^e lettre est de la forme a_{zy} avec $x < z$. Nous avons donc $u_j \prec u_{j'}$. De plus, lorsque r est une arête fille de r' , u_j est de la forme a_{xy} et $u_{j'}$ est de la forme a_{yz} . Nous avons dans ce cas aussi $u_j \prec u_{j'}$. Ceci montre que le mot $m(F)$ est F -compatible. Il apparaît donc en tant que monôme dans $r_A(\mathbf{E}_F)$. \square

Soit \leq_{lex} l'ordre lexicographique sur l'ensemble des mots sur A où l'on pose $a_{ij} \leq_{\text{lex}} a_{k\ell}$ si $i < k$, ou $i = k$ et $j \leq \ell$.

Lemme 6.3.4. *Soit F une forêt. Alors, le mot $m(F)$ est le plus petit monôme pour l'ordre \leq_{lex} qui apparaît dans $r_A(\mathbf{E}_F)$.*

Démonstration. Notons tout d'abord qu'étant donné que les monômes, vus comme des mots, qui apparaissent dans le polynôme $r_A(\mathbf{E}_F)$ sont tous de même longueur et que les lettres de A sont indexées par des entiers positifs, il existe un plus petit mot dans $r_A(\mathbf{E}_F)$ pour l'ordre \leq_{lex} .

Nous utilisons dans cette démonstration le lemme 6.3.3 de manière implicite — afin de ne pas surcharger le texte — pour s'assurer du fait que tout mot $m(F)$ apparaît bien en tant que monôme dans $r_A(\mathbf{E}_F)$.

Nous procédons par récurrence sur la taille des forêts. Si F est de taille 1, F est la forêt qui possède une unique arête. Nous avons alors $m(F) = a_{10}$, et comme a_{10} est la plus petite lettre pour l'ordre \leq_{lex} , le lemme est vérifié. Sinon, supposons que F est de taille supérieure à 2 et qu'elle est obtenue à partir d'une forêt F' par l'ajout d'une arête r_{ij} en dernière position. Nous considérons que ces forêts sont étiquetées par l'algorithme ÉTIQUETAGEMIN. Posons $u' := m(F')$. Nous avons trois cas à examiner.

Cas 1. Si l'arête r_{ij} n'est attachée à aucun des nœuds de F' , la forêt F est la concaténation des arbres de F' et de T , l'arbre à une unique arête. Étant donné que par hypothèse de récurrence, u' est le plus petit mot pour l'ordre \leq_{lex} qui soit F' -compatible, le plus petit mot qui est F -compatible est $u' \cdot a_{10}$. D'après la définition de l'application m , nous avons

$$m(F) = m(F') \cdot m(T) = u' \cdot a_{10}, \quad (6.3.24)$$

ce qui est en accord avec le lemme.

Cas 2. Si l'arête r_{ij} est attachée à un nœud y de F' , et que son extrémité x ne possède pas de frère gauche, x est le fils le plus à gauche de y . Ceci implique, par définition de l'algorithme ÉTIQUETAGEMIN, que x est étiqueté par $\mathbf{b} := \mathbf{a} + 1$ où \mathbf{a} est l'étiquette de y . Ainsi, par définition de l'application m , nous avons

$$m(F) = u' \cdot a_{\mathbf{ba}}, \quad (6.3.25)$$

et $a_{\mathbf{ba}}$ est la plus petite lettre pour l'ordre \leq_{lex} qui est en \prec -relation avec la lettre qui correspond à l'arête mère de r_{ij} . Par hypothèse de récurrence, $m(F)$ vérifie bien le lemme.

Cas 3. Autrement, l'arête r_{ij} est attachée à un nœud y de F' et possède une arête sœur gauche r_{kj} . Notons z l'extrémité de r_{kj} . D'après la définition de l'algorithme ÉTIQUETAGEMIN, x est étiqueté par $\mathbf{b} := \mathbf{a} + 1$ où \mathbf{a} est l'étiquette de z . Par définition de l'application m , nous avons

$$m(F) = u' \cdot a_{\mathbf{ba}}, \quad (6.3.26)$$

et $a_{\mathbf{ba}}$ est la plus petite lettre pour l'ordre \leq_{lex} qui est en \prec -relation avec la lettre qui correspond à r_{kj} , et, si elle existe, avec la lettre qui correspond à l'arête mère de r_{ij} . Ainsi, et par hypothèse de récurrence, $m(F)$ vérifie le lemme. \square

Lemme 6.3.5. *L'application qui à une forêt F lui associe le mot $m(F)$ est injective.*

Démonstration. Soient F_1 et F_2 deux forêts différentes et s le plus petit indice tel que la s^{e} arête r_{ij} de F_1 est différente de la s^{e} arête $r_{k\ell}$ de F_2 . Soit H la forêt constituée des $s - 1$ premières arêtes communes à F_1 et F_2 et G_1 (resp. G_2) la forêt H à laquelle on ajoute l'arête r_{ij} (resp. $r_{k\ell}$). Nous supposons que ces forêts sont étiquetées selon l'algorithme ÉTIQUETAGEMIN. Remarquons que H est de taille supérieure à 1 puisque les premières arêtes de F_1 et F_2 sont nécessairement égales. Nous avons plusieurs cas à considérer.

Cas 1. Si r_{ij} n'est pas attachée à un nœud de H et $r_{k\ell}$ n'est pas non plus attachée à un nœud de H , alors on a $G_1 = G_2$, ce qui montre que les arêtes r_{ij} et $r_{k\ell}$ sont les mêmes. Ceci est contradictoire avec nos hypothèses. Il y a donc au moins une des deux arêtes parmi r_{ij} ou $r_{k\ell}$ qui est attachée à un nœud de H .

Cas 2. Supposons sans perte de généralité que r_{ij} est attachée à un nœud y de H mais pas $r_{k\ell}$. Notons x l'extrémité de r_{ij} . Nous avons, par définition de l'application m , $m(G_1) = m(H) \cdot a_{\mathbf{ba}}$ où \mathbf{b} est l'étiquette de x et $\mathbf{b} - 1$ est l'étiquette du frère gauche de x , ou s'il n'en a pas, de son père y . Maintenant, comme H possède au moins une arête, nous avons $\mathbf{b} \geq 2$, ce qui implique $a_{\mathbf{ba}} \neq a_{10}$. Finalement, étant donné que $m(G_2) = m(H) \cdot a_{10}$ et que, par définition de l'application m , le mot $m(G_1)$ (resp. $m(G_2)$) est un préfixe de $m(F_1)$ (resp. $m(F_2)$), nous avons $m(F_1) \neq m(F_2)$.

Cas 3. Supposons que r_{ij} et $r_{k\ell}$ sont attachées respectivement à des nœuds x et y de H . Comme ces arêtes sont les plus à droite dans G_1 et G_2 , x et y figurent dans la branche la plus à droite de H . D'après l'algorithme ÉTIQUETAGEMIN, les étiquettes des nœuds sur une branche croissent strictement de la racine vers les feuilles. Ainsi, les étiquettes de x et de y sont différentes. Nous avons donc $m(G_1) = m(H) \cdot a_{\mathbf{ba}}$ où \mathbf{b} est l'étiquette de l'extrémité de r_{ij} et \mathbf{a} l'étiquette de x , et $m(G_2) = m(H) \cdot a_{\mathbf{dc}}$ où \mathbf{d} est l'étiquette de l'extrémité de $r_{k\ell}$ et \mathbf{c} l'étiquette de y . Comme $\mathbf{a} \neq \mathbf{c}$, $a_{\mathbf{ba}} \neq a_{\mathbf{dc}}$, et étant donné que $m(G_1)$ (resp. $m(G_2)$) est un préfixe de $m(F_1)$ (resp. $m(F_2)$), nous avons $m(F_1) \neq m(F_2)$. \square

Théorème 6.3.6. *Le morphisme r_A défini en (6.3.16) est une réalisation polynomiale d'algèbre de Hopf de **APE**.*

Démonstration. D'après la définition 2.1.12 du chapitre 2, nous devons montrer que, pour tout alphabet A muni de la relation \prec , r_A est un morphisme injectif d'algèbre et que l'on a l'identité

$$(r_A \otimes r_A) \Delta(\mathbf{E}_F) = r_{A \oplus A}(\mathbf{E}_F), \quad (6.3.27)$$

pour toute forêt F .

Le morphisme r_A est par définition un morphisme d'algèbre. En outre, nous avons, d'après le lemme 6.3.4 pour toute forêt F ,

$$r_A(\mathbf{E}_F) = m(F) + \sum_{\substack{u \Vdash F \\ m(F) <_{\text{lex}} u}} u. \quad (6.3.28)$$

Maintenant, étant donné que par le lemme 6.3.5, l'application $F \mapsto m(F)$ est injective, nous obtenons par triangularité le fait que r_A est un morphisme injectif.

Comme r_A est un morphisme d'algèbre, il suffit pour montrer (6.3.27) de se restreindre au cas où $F =: T$ est un arbre. Soient u un monôme qui apparaît dans $r_{A \oplus A}(\mathbf{E}_T)$, r_{ij} la s^e arête de T et $r_{k\ell}$ la t^e arête de T . Si u_s est une lettre de la seconde copie de A et que l'on a $r_{ij} \dot{\prec} r_{k\ell}$, alors, comme u est T -compatible, nous avons $u_s \prec u_t$. Ainsi, u_t est aussi une lettre qui provient de la seconde copie de A . Ceci implique que le mot u peut être identifié à un tenseur $v \otimes w$ où les lettres de v sont dans la seconde copie de A et celles de w dans la première et tel qu'il existe un coloriage admissible I tel que $T|_I = (S, R)$, v est S -compatible et w est R -compatible. D'après la proposition 6.3.2, $v \otimes w$ apparaît aussi dans $(r_A \otimes r_A) \Delta(\mathbf{E}_T)$.

Inversement, si un tenseur $v \otimes w$ apparaît dans $(r_A \otimes r_A) \Delta(\mathbf{E}_T)$, alors, par la proposition 6.3.2, il existe un coloriage admissible I de T tel que $T|_I = (S, R)$, v est S -compatible et w est R -compatible. Il suit, d'après la définition de la relation \prec que $v \otimes w$ apparaît également dans $r_{A \oplus A}(\mathbf{E}_T)$. Ceci termine de montrer (6.3.27). \square

Voici un exemple de doublement d'alphabet pour calculer le coproduit d'un élément de **APE**. Nous avons d'un côté

$$\Delta \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) = 1 \otimes \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} + \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \otimes \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} + \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \otimes \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} + \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \otimes \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} + \mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \otimes 1, \quad (6.3.29)$$

et ainsi,

$$\begin{aligned} (r_A \otimes r_A) \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) &= 1 \otimes r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) + r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) + r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) \\ &\quad + r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) + r_A \left(\mathbf{E} \begin{array}{c} \text{ } \\ \text{ } \end{array} \right) \otimes 1. \end{aligned} \quad (6.3.30)$$

De l'autre coté, en notant B la seconde copie de A nous avons

$$\begin{aligned}
& r_{A \oplus B} \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \\
&= \sum_{\substack{u_1, u_2, u_3 \in A \oplus B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 \\
&= \sum_{\substack{u_1, u_2, u_3 \in A \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_1, u_2 \in A \\ u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_1, u_3 \in A \\ u_2 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_2, u_3 \in A \\ u_1 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 \\
&\quad + \sum_{\substack{u_1 \in A \\ u_2, u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_2 \in A \\ u_1, u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_3 \in A \\ u_1, u_2 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 + \sum_{\substack{u_1, u_2, u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 \\
&= \sum_{\substack{u_1, u_2, u_3 \in A \\ u_1 \prec u_2 \\ u_3 \prec u_2}} 1 \otimes u_1 u_2 u_3 + 0 + \sum_{\substack{u_1, u_3 \in A \\ u_2 \in B}} u_2 \otimes u_1 u_3 + 0 \\
&\quad + \sum_{\substack{u_1 \in A \\ u_2, u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_2 u_3 \otimes u_1 + 0 + \sum_{\substack{u_3 \in A \\ u_1, u_2 \in B \\ u_1 \prec u_2}} u_1 u_2 \otimes u_3 + \sum_{\substack{u_1, u_2, u_3 \in B \\ u_1 \prec u_2 \\ u_3 \prec u_2}} u_1 u_2 u_3 \otimes 1 \\
&= 1 \otimes r_A \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) + r_B \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) + r_B \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \\
&\quad + r_B \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \otimes r_A \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) + r_B \left(\mathbf{E} \begin{array}{c} \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right) \otimes 1.
\end{aligned}$$

(6.3.31)

Quatrième partie

Combinatoire des arbres binaires équilibrés

Grammaires synchrones

Parce qu’elles permettent de décrire et d’engendrer des langages — ou ensembles de mots — les grammaires formelles [Cho56] sont des outils centraux en théorie des langages [Aut94]. Le processus emprunté par une grammaire formelle pour engendrer un mot est simple : on commence à partir d’un mot particulier, appelé axiome, auquel on applique successivement des dérivations, c’est-à-dire des substitutions de certains de ses facteurs par d’autres mots, selon un ensemble fixé de règles, dites règles de substitutions. Les mots que l’on peut obtenir ainsi forment le langage engendré par la grammaire. Plusieurs sortes de grammaires ont été introduites et étudiées : les grammaires hors contexte, les grammaires contextuelles, les grammaires sans restrictions, pour ne citer que ces exemples, qui possèdent chacune d’elles leur propre niveau de généralité, et permettent d’engendrer et de décrire certains types de langages plutôt que d’autres [HMU00].

De manière similaire, certains types de grammaires sont pensés pour engendrer non plus des mots mais plutôt des arbres enracinés. Ces grammaires sont connues sous le nom de grammaires d’arbres [CDG⁺07]. Pour engendrer un arbre dans une telle grammaire, on commence à partir d’un arbre particulier, également appelé axiome, auquel on applique une série de substitutions en remplaçant ses feuilles par d’autres arbres, suivant ici aussi un ensemble de règles de substitution fixé.

L’objectif de ce chapitre est d’introduire un type particulier de grammaires dans le but d’engendrer des structures arborescentes selon un tout autre processus de génération. Dans nos grammaires, les substitutions doivent être *synchrones* : le principe est que, à chaque étape de substitution, *toutes* les feuilles de l’arbre sont substituées *simultanément* par de nouveaux arbres suivant un ensemble de règles de substitution fixé. Nous qualifions naturellement ces objets de *grammaires synchrones*. La motivation principale pour l’introduction de ces structures réside dans le fait que ces grammaires permettent d’engendrer des arbres tout en conservant un contrôle sur les hauteurs de chacun de ses sous-arbres. Rappelons à ce propos que la hauteur d’un arbre est la longueur du plus long chemin connectant sa racine à l’une de ses feuilles. De plus, sous certaines conditions, nous pouvons extraire, à partir d’une grammaire synchrone, une équation fonctionnelle de point fixe pour la série génératrice qui dénombre les arbres engendrés selon leur nombre de feuilles. Les résultats présentés ici sont utilisés à plusieurs reprises dans le chapitre 8, dans le but de dénombrer plusieurs types de structures arborescentes en rapport avec les arbres binaires équilibrés et le treillis de Tamari.

Ce chapitre est organisé comme suit. Dans le paragraphe 7.1, nous introduisons les arbres à bourgeons, les grammaires synchrones, et le principe de substitution synchrone. Le paragraphe 7.2 est consacré à la description d’un procédé qui permet d’obtenir une équation fonctionnelle de point fixe pour la série génératrice dénombrant les structures arborescentes engendrées par une grammaire synchrone. Nous décrivons aussi un algorithme qui permet de calculer

ses premiers coefficients. Nous illustrons finalement dans le paragraphe 7.3, les notions présentées jusqu'alors en donnant trois exemples simples de grammaires synchrones : l'une engendre les arbres binaires parfaits, l'autre les arbres 2,3-équilibrés, et la dernière, les arbres binaires équilibrés.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir10].

7.1 Arbres à bourgeons et grammaires synchrones

7.1.1 Arbres à bourgeons

Rappelons qu'un *arbre plan* est un arbre enraciné plongé dans le plan. Autrement dit, les sous-arbres de chacun de ses nœuds sont totalement ordonnés de gauche à droite.

Définition 7.1.1. *Soit B un alphabet fini et non vide. Un arbre à bourgeons sur B — ou simplement arbre à bourgeons si le contexte est clair — est un arbre plan enraciné non vide dont les nœuds sans fils, appelés bourgeons, sont étiquetés sur B .*

Fixons pour ce chapitre un alphabet $B := \{b_1, \dots, b_k\}$ fini et non vide. Sauf mention contraire, tous les arbres à bourgeons considérés par la suite sont sur B . Soit D un arbre à bourgeons. Les *nœuds* de D sont ses sommets qui ne sont pas des bourgeons. L'ensemble des bourgeons de D est noté $\text{Brg}(D)$. Si x est un bourgeon de D étiqueté par b_i , nous dirons que x est de *type* b_i . La *frontière* de D est la suite (b_1, \dots, b_n) de ses bourgeons, lus de la gauche vers la droite. Si b est un bourgeon de D , son *évaluation* $\text{ev}(b)$ est la lettre de B qui l'étiquette. Par extension, l'*évaluation* $\text{ev}(D)$ de D est l'élément de l'algèbre commutative $\mathbb{Z}[B]$ des polynômes sur B défini par

$$\text{ev}(D) := \prod_{b \in \text{Brg}(D)} \text{ev}(b). \quad (7.1.1)$$

Nous avons ainsi par exemple

$$\text{ev} \left(\begin{array}{c} \text{blue circle} \\ | \\ \text{green circle } z \end{array} \right) = z \quad \text{et} \quad \text{ev} \left(\begin{array}{c} \text{blue circle} \\ / \quad \backslash \\ \text{green circle } x \quad \text{blue circle} \\ | \quad / \quad \backslash \\ \text{green circle } y \quad \text{green circle } x \end{array} \right) = x^2 y. \quad (7.1.2)$$

La *taille* de D est le nombre de bourgeons qu'il contient. L'ensemble des arbres à bourgeons de taille n est noté \mathcal{D}_n , et l'ensemble de tous les arbres à bourgeons est noté \mathcal{D} . Nous serons amenés à considérer des *arbres à bourgeons étiquetés*, qui sont simplement des arbres à bourgeons dont les nœuds sont étiquetés sur un alphabet A auxiliaire.

Notons que l'ensemble \mathcal{D} des arbres à bourgeons ne forme pas une classe combinatoire au sens de la définition 1.1.1 du chapitre 1. Il existe en effet pour tout $n \geq 1$ une infinité d'arbres à bourgeons de taille n . Nous pouvons par exemple construire une infinité d'arbres à bourgeons de taille 1 qui sont constitués d'une unique branche de longueur arbitrairement grande et qui possèdent un unique bourgeon.

7.1.2 Grammaires synchrones

Définitions et notations

Définition 7.1.2. *Une grammaire synchrone est un triplet $S := (B, a, R)$ où*

- B est un alphabet fini et non vide, l'alphabet des types de bourgeons ;
- a est un bourgeon étiqueté sur B , l'axiome de S ;
- $R \subseteq B \times \mathcal{D}$ est un ensemble fini tel que pour tout $b \in B$, il existe au moins un arbre à bourgeons D tel que $(b, D) \in R$. C'est l'ensemble des règles de substitution de S .

Soit $S := (B, a, R)$ une grammaire synchrone. Par souci de lisibilité, nous allons utiliser la notation suivante pour les règles de substitution. Si (\mathbf{b}, D) est une règle de substitution de S , nous la notons $\mathbf{b} \mapsto_S D$ ou simplement $\mathbf{b} \mapsto D$ si le contexte est fixé. De plus, par souci de concision, un ensemble de règles de substitution de la forme $\mathbf{b} \mapsto_S D_1, \dots, \mathbf{b} \mapsto_S D_n$ sera noté

$$\mathbf{b} \mapsto_S D_1 + \dots + D_n. \quad (7.1.3)$$

Dérivations et langages

Définition 7.1.3. Soit $S := (B, a, R)$ une grammaire synchrone et D_0 un arbre à bourgeons de frontière (b_1, \dots, b_n) où $\text{ev}(b_i) = \mathbf{b}_i$ pour tout $i \in [n]$. L'arbre à bourgeons D_1 est dérivable depuis D_0 dans S s'il existe dans R des règles de substitution $\mathbf{b}_1 \mapsto T_1, \dots, \mathbf{b}_n \mapsto T_n$ telles que, en substituant simultanément les bourgeons b_i de D_0 par la racine de T_i pour tout $i \in [n]$, on obtient D_1 . Ceci est noté $D_0 \xrightarrow{S} D_1$.

Définition 7.1.4. Un arbre à bourgeons D est engendré par une grammaire synchrone $S := (B, a, R)$ s'il existe une suite $(D_1, \dots, D_{\ell-1})$ d'arbres à bourgeons telle que

$$a \xrightarrow{S} D_1 \xrightarrow{S} \dots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D. \quad (7.1.4)$$

Nous dirons de plus que D est engendré par une dérivation de ℓ étapes et que (7.1.4) constitue une dérivation de ℓ étapes. Nous dirons aussi dans ce cas que D est engendré par S .

La grammaire synchrone S est émondée si pour tout $\mathbf{b} \in B$, il existe au moins un arbre à bourgeons D engendré par S qui contient un bourgeon de type \mathbf{b} . Dans ce qui suit, nous considérons uniquement des grammaires synchrones émondées, et de ce fait, nous ne mentionnerons plus ce qualificatif.

Définition 7.1.5. Soit $S := (B, a, R)$ une grammaire synchrone. Le langage de S d'ordre ℓ est l'élément de $\text{Vect}(\mathcal{D})$, l'espace vectoriel sur l'ensemble des arbres à bourgeons, défini par

$$\mathcal{L}_S^{(\ell)} := \sum_{a \xrightarrow{S} D_1 \xrightarrow{S} \dots \xrightarrow{S} D_\ell} D_\ell. \quad (7.1.5)$$

Par définition, le coefficient d'un arbre à bourgeons D dans $\mathcal{L}_S^{(\ell)}$ témoigne du nombre de façons d'engendrer D par des dérivations de ℓ étapes. Par extension, nous appelons *langage* de S l'ensemble \mathcal{L}_S des arbres à bourgeons engendrés par S .

Graphes de génération

Définition 7.1.6. Le graphe de génération d'ordre ℓ d'une grammaire synchrone S est le graphe orienté $\mathcal{G}_S^{(\ell)} := (V, E)$ défini par

$$V := \bigcup_{0 \leq i \leq \ell} \left\{ D : D \text{ apparaît dans } \mathcal{L}_S^{(i)} \right\}, \quad (7.1.6)$$

et

$$E := \left\{ (D_0, D_1) \in V^2 : D_0 \xrightarrow{S} D_1 \right\}. \quad (7.1.7)$$

Le graphe de génération de S est le graphe \mathcal{G}_S ainsi défini, avec $V := \mathcal{L}_S$.

Le graphe \mathcal{G}_S est potentiellement infini. Les graphes $\mathcal{G}_S^{(\ell)}$ et \mathcal{G}_S possèdent exactement une source, l'axiome a .

Un exemple

Les notions introduites dans la suite du texte seront illustrées par l'intermédiaire de la grammaire synchrone $S_{\text{ep1}} := (\{x, y\}, \textcircled{x}, R)$ où R contient les règles de substitution

$$x \mapsto \begin{array}{c} \textcircled{2} \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{y} \end{array} + \begin{array}{c} \textcircled{3} \\ \swarrow \quad \downarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{y} \quad \textcircled{x} \end{array}, \quad (7.1.8)$$

$$y \mapsto \textcircled{x}. \quad (7.1.9)$$

La Figure 7.1 montre une dérivation dans S_{ep1} .

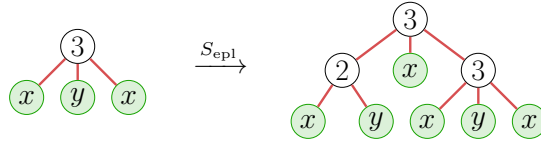


FIGURE 7.1 – Une dérivation dans la grammaire synchrone S_{ep1} .

La Figure 7.2 montre le graphe de génération d'ordre 2 de S_{ep1} .

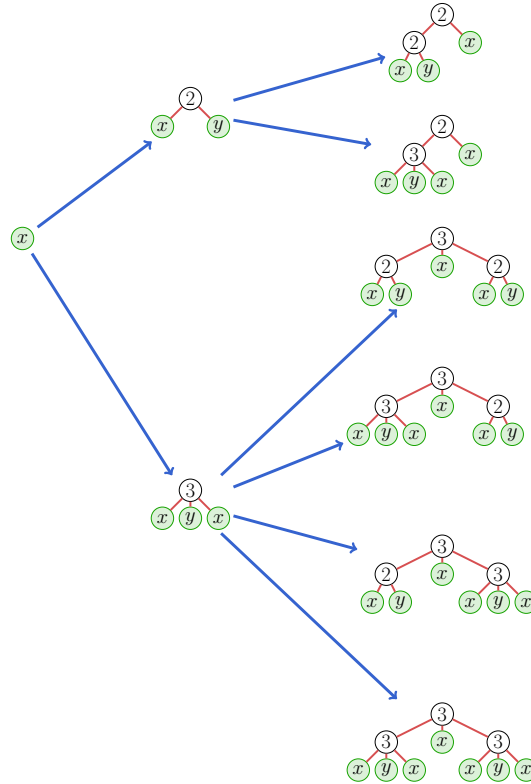


FIGURE 7.2 – Le graphe de génération d'ordre 2 de la grammaire synchrone S_{ep1} .

7.1.3 Conditions sur les grammaires synchrones

Notre attention se porte maintenant sur les grammaires synchrones qui vérifient deux propriétés bien spécifiques que nous allons définir.

Grammaires synchrones localement finies

Définition 7.1.7. Une grammaire synchrone $S := (B, a, R)$ est localement finie si pour tout $n \geq 1$, l'ensemble

$$\left\{ a \xrightarrow{S} D_1 \xrightarrow{S} \dots \xrightarrow{S} D_\ell : \ell \geq 0 \text{ et } |D_\ell| = n \right\} \quad (7.1.10)$$

est fini. En d'autres termes, une grammaire synchrone est localement finie s'il existe, pour tout $n \geq 1$, uniquement un nombre fini de dérivations qui engendrent des arbres à bourgeons de taille n .

Ainsi, et à plus forte raison, une grammaire synchrone localement finie engendre pour tout $n \geq 1$ un nombre fini d'arbres à bourgeons de taille n ainsi qu'un nombre fini d'arbres à bourgeons ayant une évaluation donnée.

Il est clair que lorsque S est localement finie, son graphe de génération \mathcal{G}_S est acyclique. En effet, par contraposée, si \mathcal{G}_S admettait un cycle, il existerait alors une infinité de dérivations qui emprunteraient ce cycle pour engendrer un même arbre à bourgeons. Remarquons cependant que la réciproque est fautive : le graphe de génération de la grammaire synchrone $S := (\{z\}, \textcircled{z}, R)$ où R contient la règle de substitution

$$z \mapsto \begin{array}{c} \textcircled{\text{blue}} \\ | \\ \textcircled{\text{green}} \end{array} z, \quad (7.1.11)$$

est constitué d'une unique branche, mais S engendre une infinité d'arbres à bourgeons différents, tous de taille 1.

Avant de donner un critère suffisant permettant d'affirmer qu'une grammaire synchrone est localement finie, rappelons ce qu'est un ordre monomial. Un *ordre monomial* est un ordre total \leq défini sur un ensemble M de monômes, et qui vérifie les deux conditions suivantes :

- (i) pour tous $x, y, z \in M$, $x \leq y$ implique $x \cdot z \leq y \cdot z$, où \cdot est le produit de monômes de M ;
- (ii) tout ensemble non vide de monômes de M possède un plus petit élément.

Un ordre total \leq_B défini sur les lettres de B donne lieu à un ordre monomial sur les monômes de $\mathbb{Z}[B]$, que nous notons aussi \leq_B par extension. En effet, si la relation \leq_B est définie sur B par $\mathbf{b}_i \leq_B \mathbf{b}_j$ pour tout $1 \leq i \leq j \leq k$, nous avons, pour tous monômes $x := \mathbf{b}_1^{\alpha_1} \dots \mathbf{b}_k^{\alpha_k}$ et $y := \mathbf{b}_1^{\beta_1} \dots \mathbf{b}_k^{\beta_k}$,

$$x \leq_B y \quad \text{si et seulement si} \quad \begin{array}{l} x = y \\ \text{ou} \\ |x| < |y| \\ \text{ou} \\ \alpha_j < \beta_j \text{ pour un } j \in [k], \text{ et } \alpha_i = \beta_i \text{ pour tout } i \in [j-1]. \end{array} \quad (7.1.12)$$

Lemme 7.1.8. Soit $S := (B, a, R)$ une grammaire synchrone. S'il existe un ordre monomial \leq_B sur les monômes de $\mathbb{Z}[B]$ tel que, pour toute dérivation dans S de la forme

$$a \xrightarrow{S} D_1 \xrightarrow{S} \dots \xrightarrow{S} D, \quad (7.1.13)$$

nous avons

$$\text{ev}(a) <_B \text{ev}(D_1) <_B \dots <_B \text{ev}(D), \quad (7.1.14)$$

alors S est localement finie.

Démonstration. Soit u un mot sur B . Étant donné que l'ensemble B des types de bourgeons est fini, et que d'après (7.1.14), chaque dérivation dans S augmente strictement l'évaluation de tout arbre à bourgeon sur lequel elle est appliquée, il existe un entier ℓ tel que toute dérivation qui engendre un arbre à bourgeon ayant u comme évaluation fasse moins de ℓ étapes. Maintenant, comme l'ensemble R des règles de substitution est fini, le nombre de dérivations qui engendrent des arbres à bourgeons ayant u comme évaluation est fini. Le résultat suit finalement du fait que comme B est fini, il existe un nombre fini d'évaluations de longueur n , et donc, également un nombre fini de dérivations qui engendrent des arbres à bourgeons de taille n . \square

Voici la condition suffisante qui permet d'affirmer qu'une grammaire synchrone est localement finie :

Lemme 7.1.9. *Soit $S := (B, a, R)$ une grammaire synchrone. S'il existe un ordre total \leq_B sur B tel que, pour toute règle de substitution $\mathbf{b} \mapsto D$ de R où D est un arbre à bourgeons de taille 1, nous avons $\mathbf{b} <_B \text{ev}(D)$, alors S est localement finie.*

Démonstration. On étend tout d'abord l'ordre total \leq_B sur B en un ordre monomial \leq_B sur l'ensemble des monômes de $\mathbb{Z}[B]$ suivant (7.1.12).

Considérons maintenant un arbre à bourgeons D_0 engendré par S et un arbre à bourgeons D_1 dérivable depuis D_0 . Montrons que l'on a bien $\text{ev}(D_0) <_B \text{ev}(D_1)$. Nous avons pour cela deux cas à considérer.

Cas 1. S'il existe au moins un bourgeon de D_0 qui est substitué par un arbre à bourgeons de taille supérieure à 2, alors $|\text{ev}(D_0)| < |\text{ev}(D_1)|$ et par conséquent, $\text{ev}(D_0) <_B \text{ev}(D_1)$.

Cas 2. Sinon, D_0 et D_1 sont de la même taille. Par hypothèse, tous les bourgeons de la frontière (b_1, \dots, b_n) de D_0 sont substitués par n arbres à bourgeons de taille 1 contenant les bourgeons c_1, \dots, c_n tels que $\text{ev}(b_i) <_B \text{ev}(c_i)$ pour tout $i \in [n]$. Ainsi, nous avons $\text{ev}(D_0) <_B \text{ev}(D_1)$.

Le résultat suit finalement du lemme 7.1.8. \square

À titre d'exemple, la grammaire synchrone S_{ep1} , en ordonnant son alphabet de bourgeons $B := \{x, y\}$ par $y <_B x$, vérifie la condition du lemme 7.1.9 et est ainsi localement finie.

Grammaires synchrones non ambiguës

Définition 7.1.10. *Une grammaire synchrone S est non ambiguë si pour tout arbre à bourgeons D , il existe au plus un entier positif ℓ et une suite $(D_1, \dots, D_{\ell-1})$ telle que (7.1.4) est établie.*

Étant donné qu'il existe par définition dans le graphe de génération \mathcal{G}_S d'une grammaire synchrone non ambiguë S exactement un chemin qui connecte l'axiome de S à tout arbre à bourgeons engendré par S , \mathcal{G}_S est un arbre enraciné en son axiome. De plus, la réciproque est vraie : toute grammaire synchrone dont le graphe de génération est un arbre est d'emblée non ambiguë.

Nous donnons ici un critère suffisant pour affirmer qu'une grammaire synchrone localement finie est non ambiguë.

Lemme 7.1.11. *Soit $S := (B, a, R)$ une grammaire synchrone localement finie. Si pour tout $\mathbf{b} \in B$ et pour toutes règles de substitution $\mathbf{b} \mapsto T_0$ et $\mathbf{b} \mapsto T_1$ de R où $T_0 \neq T_1$, il existe deux nœuds différents dans T_0 et T_1 qui se trouvent aux mêmes positions dans leurs arbres respectifs, alors, S est non ambiguë.*

Démonstration. Soit D un arbre à bourgeons engendré par S , et D_0 et D_1 deux arbres à bourgeons différents et dérivables depuis D . Parmi d'autres substitutions, D_0 (resp. D_1) est obtenu en substituant un de ses bourgeons par un arbre à bourgeons T_0 (resp. T_1), et par hypothèse,

il y a aux mêmes positions dans T_0 et T_1 deux nœuds différents. Ceci montre que tout arbre à bourgeons obtenu en réalisant une suite de dérivations depuis D_0 et tout arbre à bourgeons obtenu en réalisant une suite de dérivations depuis D_1 sont différents puisqu'il existe dans ces arbres deux nœuds différents aux mêmes positions. De plus, comme S est localement finie, son graphe de génération est acyclique, ce qui montre que S est non ambiguë. \square

À titre d'exemple, le lemme 7.1.11 montre que la grammaire S_{epi} est non ambiguë. En effet, celle-ci est à la fois localement finie et de plus, seuls les bourgeons de type x peuvent être substitués par des arbres à bourgeons différents et chacun d'eux possède une racine différente : l'une est d'arité 2 tandis que l'autre est d'arité 3.

7.2 Grammaires synchrones et dénombrement

La principale raison qui motive l'introduction des grammaires synchrones réside dans le fait qu'elles constituent un nouvel outil pour dénombrer des familles d'objets combinatoires. Dans cet objectif, nous montrons dans ce paragraphe comment obtenir, à partir d'une grammaire synchrone localement finie et non ambiguë, une série génératrice qui dénombre selon leur évaluation les arbres à bourgeons que celle-ci engendre.

Les séries génératrices obtenues à partir des grammaires synchrones s'expriment en réalité comme solutions d'équations fonctionnelles de point fixe. Le calcul des coefficients de ces séries se fait par *itération*. Nous proposons de ce fait un procédé qui permet de les calculer.

Nous terminons ce paragraphe en illustrant le fait que les grammaires synchrones sont des objets qui se prêtent particulièrement bien à des spécialisations et à des raffinements dans les objets qu'elles dénombrent. Il est ainsi possible de dénombrer une famille d'objets selon des statistiques.

7.2.1 Série génératrice d'une grammaire synchrone

Définition 7.2.1. Soit $S := (B, a, R)$ une grammaire synchrone. La série génératrice d'ordre ℓ de S , notée $\mathcal{S}_S^{(\ell)}$, est le polynôme de $\mathbb{Z}[B]$ défini par

$$\mathcal{S}_S^{(\ell)}(\mathbf{b}_1, \dots, \mathbf{b}_k) := \text{ev} \left(\mathcal{L}_S^{(\ell)} \right), \quad (7.2.1)$$

où le symbole ev du membre droit de (7.2.1) désigne l'évaluation sur les arbres à bourgeons, étendue par linéarité. De plus, dans le cas où S est localement finie, la série génératrice de S , notée \mathcal{S}_S , est l'élément de $\mathbb{Z}[[B]]$ défini par

$$\mathcal{S}_S(\mathbf{b}_1, \dots, \mathbf{b}_k) := \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\mathbf{b}_1, \dots, \mathbf{b}_k). \quad (7.2.2)$$

Soit $S := (B, a, R)$ une grammaire synchrone localement finie. Comme mentionné précédemment, le fait que S est localement finie implique que le nombre d'arbres à bourgeons engendrés par S ayant une évaluation donnée est fini. Ceci montre que la série \mathcal{S}_S est bien définie, *i.e.*, tous les coefficients de \mathcal{S}_S sont dans \mathbb{N} . De plus, remarquons que si S est également non ambiguë, alors pour tout monôme $u := \mathbf{b}_1^{\alpha_1} \dots \mathbf{b}_k^{\alpha_k}$, le coefficient $[u]\mathcal{S}_S$ est le nombre d'arbres à bourgeons engendrés par S qui ont u comme évaluation, *i.e.*, une frontière constituée de α_i occurrences de bourgeons de type \mathbf{b}_i , pour tout $i \in [k]$.

Par exemple, les premières séries génératrices d'ordre ℓ de S_{epI} sont

$$\mathcal{S}_{S_{\text{epI}}}^{(0)}(x, y) = x, \quad (7.2.3)$$

$$\mathcal{S}_{S_{\text{epI}}}^{(1)}(x, y) = xy + x^2y, \quad (7.2.4)$$

$$\mathcal{S}_{S_{\text{epI}}}^{(2)}(x, y) = x^2y + x^3y + x^3y^2 + 2x^4y^2 + x^5y^2. \quad (7.2.5)$$

et sa série génératrice $\mathcal{S}_{S_{\text{epI}}}$ est de la forme

$$\mathcal{S}_{S_{\text{epI}}}(x, y) = x + xy + 2x^2y + x^3y + x^3y^2 + 2x^4y^2 + x^5y^2 + \dots \quad (7.2.6)$$

7.2.2 Calculer la série génératrice d'une grammaire synchrone

Soit $S := (B, a, R)$ une grammaire synchrone. Posons maintenant la notation suivante. Pour tout $\mathbf{b} \in B$, $\text{subs}(\mathbf{b})$ désigne le polynôme de $\mathbb{Z}[B]$ défini par

$$\text{subs}(\mathbf{b}) := \sum_{(\mathbf{b}, D) \in R} \text{ev}(D). \quad (7.2.7)$$

Nous avons par exemple pour S_{epI} ,

$$\text{subs}(x) = xy + x^2y, \quad (7.2.8)$$

$$\text{subs}(y) = x. \quad (7.2.9)$$

Le lemme suivant donne un moyen de calculer la série génératrice d'une grammaire synchrone d'ordre ℓ par *itération*. En d'autres termes, le procédé décrit ici permet le calcul de la série génératrice d'ordre $\ell + 1$ en se basant sur la série génératrice d'ordre ℓ .

Lemme 7.2.2. *Soit $S := (B, a, R)$ une grammaire synchrone. Pour tout $\ell \geq 0$, $\mathcal{S}_S^{(\ell)}$ vérifie*

$$\mathcal{S}_S^{(\ell)}(\mathbf{b}_1, \dots, \mathbf{b}_k) = \begin{cases} \text{ev}(a) & \text{si } \ell = 0, \\ \mathcal{S}_S^{(\ell-1)}(\text{subs}(\mathbf{b}_1), \dots, \text{subs}(\mathbf{b}_k)) & \text{sinon.} \end{cases} \quad (7.2.10)$$

Démonstration. Si $\ell = 0$, le seul arbre à bourgeons engendré par une dérivation de zéro étapes dans S est son axiome a . De ce fait, le lemme est correct dans ce cas.

Soit $\ell \geq 1$ et supposons qu'il existe la suite de dérivations suivante dans S :

$$a \xrightarrow{S} D_1 \xrightarrow{S} \dots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D_\ell. \quad (7.2.11)$$

Soit n la taille de l'arbre à bourgeons $D_{\ell-1}$. Par définition, D_ℓ est obtenu en substituant les bourgeons b_i de $D_{\ell-1}$ par des arbres à bourgeons T_i pour tout $i \in [n]$. Du point de vue des polynômes, le monôme $\text{ev}(D_\ell)$ est obtenu par les substitutions polynomiales $\text{ev}(b_i) \leftarrow \text{ev}(T_i)$ dans $\mathcal{S}_S^{(\ell-1)}$. Par conséquent, le polynôme $\mathcal{S}_S^{(\ell)}$ est obtenu à partir de $\mathcal{S}_S^{(\ell-1)}$ en réalisant la substitution polynomiale $\mathbf{b} \leftarrow \text{subs}(\mathbf{b})$ pour tout $\mathbf{b} \in B$, ce qui montre (7.2.10). \square

Proposition 7.2.3. *Soit $S := (B, a, R)$ une grammaire synchrone localement finie. La série génératrice \mathcal{S}_S vérifie l'équation fonctionnelle de point fixe*

$$\mathcal{S}_S(\mathbf{b}_1, \dots, \mathbf{b}_k) = \text{ev}(a) + \mathcal{S}_S(\text{subs}(\mathbf{b}_1), \dots, \text{subs}(\mathbf{b}_k)). \quad (7.2.12)$$

Démonstration. En utilisant le lemme 7.2.2, nous obtenons

$$\mathcal{S}_S(\mathbf{b}_1, \dots, \mathbf{b}_k) = \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\mathbf{b}_1, \dots, \mathbf{b}_k) \quad (7.2.13)$$

$$= \mathcal{S}_S^{(0)}(\mathbf{b}_1, \dots, \mathbf{b}_k) + \sum_{\ell \geq 1} \mathcal{S}_S^{(\ell)}(\mathbf{b}_1, \dots, \mathbf{b}_k) \quad (7.2.14)$$

$$= \text{ev}(a) + \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell+1)}(\mathbf{b}_1, \dots, \mathbf{b}_k) \quad (7.2.15)$$

$$= \text{ev}(a) + \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\text{subs}(\mathbf{b}_1), \dots, \text{subs}(\mathbf{b}_k)) \quad (7.2.16)$$

$$= \text{ev}(a) + \mathcal{S}_S(\text{subs}(\mathbf{b}_1), \dots, \text{subs}(\mathbf{b}_k)). \quad \square$$

La proposition 7.2.3 donne une équation fonctionnelle de point fixe pour la série génératrice d'une grammaire synchrone S localement finie, et le lemme 7.2.2 donne, quant à lui, un moyen de l'approximer jusqu'à un ordre n désiré en sommant les polynômes $\mathcal{S}_S^{(\ell)}$ pour tout $0 \leq \ell \leq n$.

Dans notre exemple, la série génératrice de S_{epi} vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\text{epi}}}(x, y) = x + \mathcal{S}_{S_{\text{epi}}}(xy + x^2y, x). \quad (7.2.17)$$

7.2.3 Spécialisations et raffinements

Dans bon nombre de situations, certains bourgeons d'une grammaire synchrone sont nécessaires uniquement dans un rôle de *catalyseur*, *i.e.*, les objets que l'on souhaite dénombrer sont ceux exempts de ces bourgeons, ces derniers étant uniquement utiles dans le processus de dérivation. Il est utile dans ce cas de spécialiser à 0 une variable \mathbf{b} qui joue un rôle de catalyseur dans la série génératrice \mathcal{S}_S . Ainsi, les monômes qui correspondent à cette variable sont annihilés, et les arbres à bourgeons qui possèdent des bourgeons de type \mathbf{b} ne sont de ce fait pas dénombrés.

De même, on peut souhaiter dénombrer les arbres à bourgeons engendrés par S non plus selon leur évaluation, mais plus grossièrement selon leur taille. Dans ce cas, la spécialisation à t de chacune des variables \mathbf{b} où t est un paramètre formel convient. De plus, il est facile d'accorder des tailles différentes à chacun des types de bourgeons qui constituent les arbres à bourgeons du langage de S en spécialisant chacune des variables \mathbf{b} par t^n où n est la taille des bourgeons de type \mathbf{b} .

Un procédé similaire permet de raffiner \mathcal{S}_S en tenant compte du nombre d'applications de certaines règles de substitution qui interviennent pour engendrer les arbres à bourgeons du langage de S . En effet, pour tenir compte du nombre d'applications d'une règle de substitution $\mathbf{b} \mapsto D$ intervenant dans la génération d'un arbre à bourgeons, il suffit de poser

$$\text{subs}_\xi(\mathbf{b}) := \text{subs}(\mathbf{b}) + (\xi - 1) \text{ev}(D), \quad (7.2.18)$$

où ξ est un paramètre formel, et d'utiliser $\text{subs}_\xi(\mathbf{b})$ à la place de $\text{subs}(\mathbf{b})$ dans l'expression de la série génératrice de S . Ainsi, le paramètre ξ compte le nombre d'applications de la règle de substitution $\mathbf{b} \mapsto D$. De cette manière, une grammaire synchrone permet de dénombrer les éléments de son langage suivant des statistiques.

Par exemple, en ce qui concerne la grammaire synchrone S_{epi} , pour tenir compte du nombre d'applications de la règle de substitution

$$x \mapsto \begin{array}{c} \textcircled{3} \\ \swarrow \quad \downarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{y} \quad \textcircled{x} \end{array}, \quad (7.2.19)$$

il suffit de poser

$$\text{subs}_\xi(x) := xy + x^2y\xi, \quad (7.2.20)$$

de sorte que la série génératrice $\mathcal{S}_{S_{\text{epI}}}$, vérifiant l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\text{epI}}}(x, y, \xi) = x + \mathcal{S}_{S_{\text{epI}}}(xy + x^2y\xi, x, \xi), \quad (7.2.21)$$

dénombrer les arbres à bourgeons engendrés par S_{epI} selon la statistique associant à un arbre son nombre de nœuds d'arité 3.

7.3 Exemples de grammaires synchrones

Nous considérons ici trois exemples de grammaires synchrones pour illustrer les concepts présentés dans les paragraphes précédents.

7.3.1 La grammaire synchrone des arbres binaires parfaits

Soit la grammaire synchrone $S_{\text{perf}} := (\{x\}, \textcircled{x}, R)$ où R contient l'unique règle de substitution

$$x \mapsto \begin{array}{c} \bigcirc \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{x} \end{array} . \quad (7.3.1)$$

En identifiant les bourgeons de type x avec des feuilles, le langage $\mathcal{L}_{S_{\text{perf}}}$ est l'ensemble des *arbres binaires parfaits*, c'est-à-dire l'ensemble des arbres binaires qui apparaissent dans la suite $(T_i)_{i \geq 0}$ définie par $T_0 := \perp$ et $T_{i+1} := T_i \wedge T_i$.

Cette grammaire synchrone est localement finie car la taille d'un arbre à bourgeons engendré par S_{perf} augmente strictement lors de chaque étape de dérivation. De plus, comme S_{perf} est localement finie et que R contient une seule règle de substitution, le graphe de génération $\mathcal{G}_{S_{\text{perf}}}$ contient un unique chemin de longueur maximale, ce qui montre que S_{perf} est non ambiguë. Ainsi, la série $\mathcal{S}_{S_{\text{perf}}}$ est bien définie, et par la proposition 7.2.3, celle-ci vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\text{perf}}}(x) = x + \mathcal{S}_{S_{\text{perf}}}(x^2), \quad (7.3.2)$$

et dénombre les arbres binaires parfaits selon leur nombre de feuilles.

Les premiers polynômes $\mathcal{S}_{S_{\text{perf}}}^{(\ell)}$ sont

$$\mathcal{S}_{S_{\text{perf}}}^{(0)}(x) = x, \quad \mathcal{S}_{S_{\text{perf}}}^{(1)}(x) = x^2, \quad \mathcal{S}_{S_{\text{perf}}}^{(2)}(x) = x^4, \quad \mathcal{S}_{S_{\text{perf}}}^{(3)}(x) = x^8, \quad (7.3.3)$$

et sa série génératrice est de la forme

$$\mathcal{S}_{S_{\text{perf}}}(x) = \sum_{n \geq 0} x^{2^n} = x + x^2 + x^4 + x^8 + \dots . \quad (7.3.4)$$

7.3.2 La grammaire synchrone des arbres 2,3-équilibrés

Soit la grammaire synchrone $S_{23} := (\{x\}, \textcircled{x}, R)$ où R contient les règles de substitution

$$x \mapsto \begin{array}{c} \textcircled{2} \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{x} \end{array} + \begin{array}{c} \textcircled{3} \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{x} \quad \textcircled{x} \end{array} . \quad (7.3.5)$$

En identifiant les bourgeons de type x avec des feuilles, le langage de S_{23} est l'ensemble des *arbres 2,3-équilibrés*, qui sont des arbres plans enracinés tels que tout nœud interne possède

exactement 2 ou 3 fils, et tous les chemins connectant la racine aux feuilles sont de mêmes longueurs (voir [Odl82], [FS09]).

Étant donné que chaque étape de dérivation augmente strictement la taille de l'arbre à bourgeons sur lequel elle est appliquée, S_{23} est localement finie. De plus, S_{23} vérifie les hypothèses du lemme 7.1.11 et est ainsi non ambiguë. En effet, les deux arbres à bourgeons qui apparaissent dans les deux règles de substitution ont des racines différentes : l'une d'elles est d'arité 2 tandis que l'autre est d'arité 3. La série $\mathcal{S}_{S_{23}}$ est donc bien définie, et par la proposition 7.2.3, celle-ci vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{23}}(x) = x + \mathcal{S}_{S_{23}}(x^2 + x^3), \quad (7.3.6)$$

et dénombre les arbres 2,3-équilibrés selon leur nombre de feuilles.

Les premiers polynômes $\mathcal{S}_{S_{23}}^{(\ell)}$ sont

$$\mathcal{S}_{S_{23}}^{(0)}(x) = x, \quad (7.3.7)$$

$$\mathcal{S}_{S_{23}}^{(1)}(x) = x^2 + x^3, \quad (7.3.8)$$

$$\mathcal{S}_{S_{23}}^{(2)}(x) = x^4 + 2x^5 + 2x^6 + 3x^7 + 3x^8 + x^9. \quad (7.3.9)$$

Une interprétation du polynôme $\mathcal{S}_{S_{23}}^{(2)}(x)$ est la suivante : par des dérivations de deux étapes depuis son axiome, S_{23} engendre un arbre à bourgeons de taille 4, deux arbres à bourgeons de taille 5, deux arbres à bourgeons de taille 6, trois arbres à bourgeons de taille 7, trois arbres à bourgeons de taille 8, et un arbre à bourgeons de taille 9.

7.3.3 La grammaire synchrone des arbres binaires équilibrés

Considérons maintenant la grammaire synchrone $S_{\text{bal}} := (\{x, y\}, \textcircled{x}, R)$ où R contient les règles de substitution

$$x \mapsto \begin{array}{c} \textcircled{-1} \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{y} \end{array} + \begin{array}{c} \textcircled{0} \\ \swarrow \quad \searrow \\ \textcircled{x} \quad \textcircled{x} \end{array} + \begin{array}{c} \textcircled{1} \\ \swarrow \quad \searrow \\ \textcircled{y} \quad \textcircled{x} \end{array}, \quad (7.3.10)$$

$$y \mapsto \textcircled{x}. \quad (7.3.11)$$

Comme nous allons le voir, en annihilant les arbres à bourgeons qui contiennent des bourgeons de type y — qui jouent le rôle de catalyseurs — et en identifiant les bourgeons de type x avec des feuilles, le langage S_{bal} est l'ensemble des arbres binaires équilibrés (voir la définition 8.1.1 du chapitre 8).

La *mesure de déséquilibre* d'un nœud d'un arbre binaire est la différence de hauteur entre son sous-arbre droit et son sous-arbre gauche.

Proposition 7.3.1. *Soit D un arbre à bourgeons engendré par S_{bal} contenant uniquement des bourgeons de type x . Alors, les nœuds de D sont étiquetés par leur mesure de déséquilibre.*

Démonstration. Toute étape de dérivation menant à la génération de D substitue chaque bourgeon de type x par un arbre à bourgeons de hauteur 2, et chaque bourgeon de type y par un arbre à bourgeons de hauteur 1. Par conséquent, toute étape de dérivation incrémente la hauteur des sous-arbres qui contiennent un bourgeon de type x , et le rôle des bourgeons de type y est d'imposer un délai, pendant une étape de dérivation, sur la croissance de la branche qui le contient. Ainsi, des nœuds ayant des mesures de déséquilibre à -1 et 1 peuvent être créés. Finalement, comme D ne possède aucun bourgeon de type y , chaque délai de croissance est respecté, et ainsi, les étiquettes de D sont les mesures de déséquilibre de ses nœuds. \square

La proposition 7.3.1 montre que les arbres à bourgeons engendrés par S_{bal} contenant uniquement des bourgeons de type x sont des arbres binaires équilibrés. De plus, il suit par induction structurelle sur l'ensemble des arbres binaires équilibrés que tout arbre binaire équilibré peut être engendré par S_{bal} . En effet, l'arbre vide peut être engendré puisqu'il correspond à son axiome, et si T est un arbre binaire équilibré et z sa racine, par hypothèse d'induction, son sous-arbre gauche et son sous-arbre droit peuvent être engendrés par S_{bal} . Pour engendrer T , il suffit de réaliser la première étape de substitution suivant la mesure de déséquilibre de z . La Figure 7.3 montre les étapes de dérivation qui engendrent un arbre binaire équilibré.

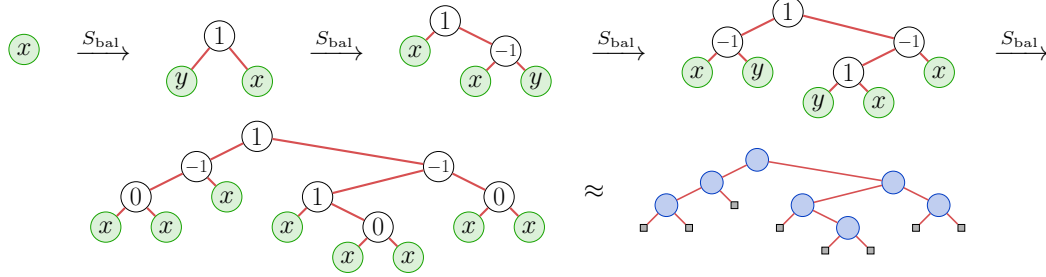


FIGURE 7.3 – Les étapes de dérivation pour la génération d'un arbre binaire équilibré par la grammaire synchrone S_{bal} .

En posant $y \leq_B x$, S_{bal} vérifie les conditions du lemme 7.1.9, ce qui montre qu'elle est localement finie. De plus, le lemme 7.1.11 montre que S_{bal} est non ambiguë puisque les arbres à bourgeons qui apparaissent dans ses règles de substitution possèdent une racine différente puisque leur étiquetage diffère. La proposition 7.3.1 montre que l'étiquetage des nœuds est consistant. Ainsi, la série génératrice $\mathcal{S}_{S_{\text{bal}}}$ est bien définie. Par la proposition 7.2.3, la série génératrice qui dénombre les éléments de $\mathcal{L}_{S_{\text{bal}}}$ vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\text{bal}}}(x, y) = x + \mathcal{S}_{S_{\text{bal}}}(x^2 + 2xy, x). \quad (7.3.12)$$

Les premiers polynômes $\mathcal{S}_{S_{\text{bal}}}^{(\ell)}$ sont

$$\mathcal{S}_{S_{\text{bal}}}^{(0)}(x, y) = x, \quad (7.3.13)$$

$$\mathcal{S}_{S_{\text{bal}}}^{(1)}(x, y) = 2xy + x^2, \quad (7.3.14)$$

$$\mathcal{S}_{S_{\text{bal}}}^{(2)}(x, y) = 4x^2y + 2x^3 + 4x^2y^2 + 4x^3y + x^4. \quad (7.3.15)$$

Comme nous l'avons mentionné, pour dénombrer les arbres binaires équilibrés, il faut rejeter les éléments de $\mathcal{L}_{S_{\text{bal}}}$ qui contiennent des bourgeons de type y . Pour cette raison, la série génératrice qui dénombre les arbres binaires équilibrés selon leur nombre de feuilles est donnée par la spécialisation $\mathcal{S}_{S_{\text{bal}}}(x, 0)$. Notons que cette équation fonctionnelle de point fixe est obtenue également dans [BLL88], [BLL94], et [Knu98] dans un autre formalisme et par d'autres méthodes.

Arbres binaires équilibrés et treillis de Tamari

Les arbres binaires de recherche sont des structures de données adaptées pour représenter des ensembles dynamiques totalement ordonnés (voir [AU94], [Knu98], [CLRS03]). Les algorithmes qui résolvent la plupart des questions sur les ensembles dynamiques, comme l'insertion, la suppression, ou la recherche d'un élément donné, consomment un temps linéaire en fonction de la hauteur de l'arbre binaire qui l'encode, et si cet arbre binaire est *équilibré*, ces opérations sont réalisables en temps logarithmique en fonction du cardinal de l'ensemble représenté. Rappelons qu'un arbre binaire est équilibré si pour chacun de ses nœuds x , les hauteurs du sous-arbre gauche et du sous-arbre droit de x diffèrent d'au plus un.

L'algorithmique des arbres binaires équilibrés repose essentiellement sur l'opération de *rotation* (la définition de cette opération est rappelée dans le paragraphe 1.3.3 du chapitre 1). Une insertion ou une suppression d'un élément dans un ensemble dynamique totalement ordonné modifie l'arbre binaire qui l'encode et peut dans certains cas le déséquilibrer. L'efficacité de ces algorithmes tient au fait, comme l'on montré Adelson-Velsky et Landis [AVL62], qu'un arbre binaire de recherche ainsi modifié demande au plus deux rotations pour se rééquilibrer, et ce, quelle que soit sa taille.

De manière surprenante, cette opération apparaît dans un contexte différent puisqu'elle permet de définir un ordre partiel sur l'ensemble des arbres binaires d'une taille fixée. Un arbre binaire T_0 est inférieur à un arbre binaire T_1 s'il est possible d'obtenir T_1 en appliquant une ou plusieurs rotations orientées dans T_0 . Cet ordre partiel, connu sous le nom d'*ordre de Tamari* [Tam62], [Sta99a], [Knu04], définit également une structure de treillis sur l'ensemble des arbres binaires d'une taille fixée (la définition précise du treillis de Tamari est rappelée dans le paragraphe 1.3.3 du chapitre 1).

Étant donné que d'un côté, l'équilibre d'un arbre binaire est maintenu par l'intermédiaire de rotations, et que d'un autre, les arbres binaires sont munis naturellement d'une structure d'ordre partiel induite précisément par les rotations, nous nous proposons dans ce chapitre d'étudier la question de savoir si les arbres binaires équilibrés jouent un rôle particulier dans le treillis de Tamari. Notre objectif est de combiner ces deux points de vue sur l'opération de rotation : le premier étant algorithmique, et le second, combinatoire. Des expérimentations sur les premiers treillis de Tamari montrent que les intervalles $[T, T']$, où T et T' sont des arbres binaires équilibrés, sont uniquement constitués d'arbres binaires équilibrés. La démonstration de cette propriété est l'un des résultats principaux de ce chapitre. Comme conséquence, nous donnons une caractérisation de la forme de ces intervalles, et, en utilisant notre notion de grammaire synchrone introduite dans le chapitre 7, nous les dénombrons.

Une autre motivation de ce travail, plus algébrique, est justifiée par l'observation suivante. Le produit de deux éléments de la base fondamentale de l'algèbre de Hopf **PBT** de Loday-Ronco des arbres binaires [LR98], [HNT05] (voir aussi le paragraphe 5.1.3 du chapitre 5) peut s'interpréter comme un intervalle particulier du treillis de Tamari [LR02]. Ainsi, étant donné que l'ensemble des arbres binaires équilibrés est un sous-ensemble de l'ensemble des arbres binaires, et que, en vertu de l'expérimentation précédente, l'ensemble des arbres binaires équilibrés est clos par intervalle dans le treillis de Tamari, il semble naturel de construire une sous-algèbre de Hopf de **PBT** basée sur les arbres binaires équilibrés. Malheureusement, aucun résultat dans cette direction n'a été obtenu.

Ce chapitre est organisé de la manière suivante. Nous commençons le paragraphe 8.1 par des rappels autour de la notion d'équilibre dans un arbre binaire. Nous établissons ensuite les outils en vue montrer que l'ensemble des arbres binaires équilibrés est clos par intervalle dans le treillis de Tamari. Nous introduisons ensuite dans le paragraphe 8.2 une notion de motif d'arbre binaire, que nous nommons *motif de déséquilibre*, ainsi qu'une notion d'évitement de motif. Une partition de l'ensemble des arbres binaires équilibrés en trois sous-ensembles suivant la position de leurs éléments dans le treillis de Tamari est proposée, et nous montrons ensuite que les éléments de ces ensembles peuvent être décrits comme les arbres binaires équilibrés qui évitent un certain ensemble de motifs de déséquilibre. Nous construisons également une grammaire synchrone qui engendre les éléments de l'un de ces trois ensembles et obtenons une équation fonctionnelle de point fixe pour la série génératrice qui dénombre ses éléments. Dans le paragraphe 8.3, nous regardons les intervalles d'arbres binaires équilibrés de plus près, et montrons qu'ils sont isomorphes, en tant que posets, à des hypercubes. De plus, en encodant les intervalles d'arbres binaires équilibrés par des structures arborescentes particulières, nous construisons une grammaire synchrone qui les engendre et obtenons par conséquent une équation fonctionnelle de point fixe pour la série génératrice des intervalles d'arbres binaires équilibrés. Nous procédons ensuite de même pour les intervalles maximaux d'arbres binaires équilibrés. Finalement, nous considérons dans le paragraphe 8.4 une généralisation des arbres binaires équilibrés et montrons entre autre que l'ensemble des arbres binaires équilibrés est le seul ensemble parmi cette généralisation qui soit à la fois clos par intervalle dans le treillis de Tamari et tel que l'ordre de Tamari réduit à ses éléments possède des intervalles non triviaux. Nous terminons en nous intéressant à trois autres familles d'arbres binaires qui sont closes par intervalle dans le treillis de Tamari, à savoir, les arbres binaires équilibrés en taille, les arbres binaires ayant une canopée fixée et les arbres binaires ayant un indice de Narayana fixé.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir10].

8.1 Clôture par intervalles des arbres binaires équilibrés

8.1.1 Arbres binaires équilibrés

Si T est un arbre binaire, $h(T)$ désigne sa *hauteur*, qui est la longueur du plus long chemin qui connecte sa racine à l'une de ses feuilles. Plus formellement,

$$h(T) := \begin{cases} 1 + \max\{h(L), h(R)\} & \text{si } T = L \wedge R, \\ 0 & \text{sinon } (T = \perp). \end{cases} \quad (8.1.1)$$

Par exemple, nous avons $h(\perp) = 0$, $h(\text{arbre à gauche}) = 1$, et $h(\text{arbre à droite}) = 2$.

L'application d_T associe à chaque nœud x de T un élément de \mathbb{Z} , à savoir, sa *mesure de déséquilibre*, définie par

$$d_T(x) := h(R) - h(L), \quad (8.1.2)$$

où L (resp. R) est le sous-arbre gauche (resp. droit) de x . La figure 8.1 montre un exemple d'arbre binaire étiqueté par les mesures de déséquilibre de ses nœuds.

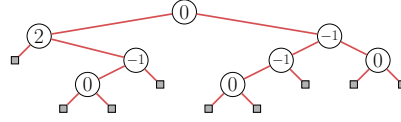


FIGURE 8.1 – Un arbre binaire étiqueté par les mesures de déséquilibre de chacun de ses nœuds.

Un nœud x est *équilibré* si

$$d_T(x) \in \{-1, 0, 1\}. \quad (8.1.3)$$

Les *arbres binaires équilibrés* forment un sous-ensemble de \mathcal{T} constitué des arbres binaires qui ont la propriété d'être équilibrés :

Définition 8.1.1. *Un arbre binaire est équilibré si tous ses nœuds sont équilibrés. La classe combinatoire des arbres binaires équilibrés est notée \mathcal{E} . La taille d'un arbre binaire équilibré est son nombre de nœuds.*

Les cardinaux des ensembles \mathcal{E}_n constituent la suite **A006265** de [Slo] qui commence par

$$1, 1, 2, 1, 4, 6, 4, 17, 32, 44, 60, 70, 184, 476, 872, 1553, 2720, 4288, 6312, 9004. \quad (8.1.4)$$

La figure 8.2 représente les éléments des ensembles \mathcal{E}_n pour $0 \leq n \leq 6$.

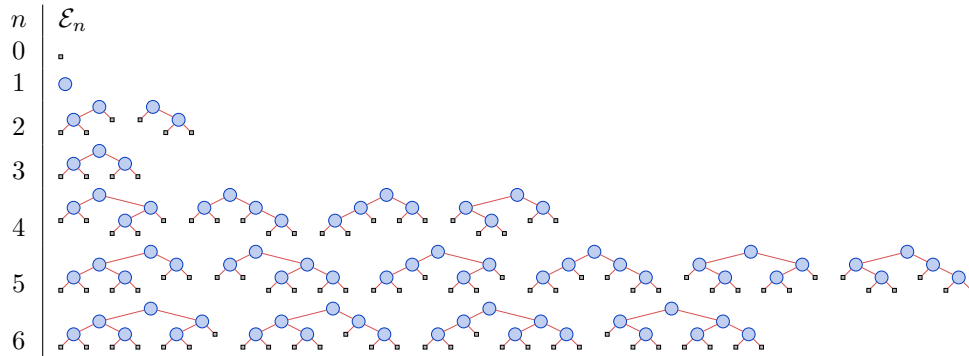


FIGURE 8.2 – Les arbres binaires équilibrés de taille inférieure à 6.

8.1.2 Rotations dans un arbre binaire équilibré

Commençons par nous intéresser aux modifications des mesures de déséquilibre des nœuds d'un arbre binaire équilibré de la forme $T_0 := (A \wedge B) \wedge C$ lorsqu'une rotation en sa racine lui est appliquée. Soit T_1 l'arbre binaire ainsi obtenu, y la racine de T_0 , et x le fils gauche de y dans T_0 (pour une illustration de cette situation, regarder la figure 1.4 du chapitre 1 en considérant que y est la racine de T_0 et x la racine de T_1). Remarquons en premier lieu que les mesures de déséquilibre des nœuds des sous-arbres A , B et C ne changent pas après cette rotation. En effet, seules les mesures de déséquilibre de x et de y sont éventuellement modifiées. Comme T_0 est équilibré, nous avons $d_{T_0}(x) \in \{-1, 0, 1\}$ et $d_{T_0}(y) \in \{-1, 0, 1\}$. Par conséquent, le couple $(d_{T_0}(x), d_{T_0}(y))$ peut prendre neuf valeurs différentes.

Voici la liste des mesures de déséquilibre de x et de y dans T_0 et T_1 , exprimée sous la forme $(d_{T_0}(x), d_{T_0}(y)) \longrightarrow (d_{T_1}(x), d_{T_1}(y))$:

$$\begin{array}{lll}
(\text{R1}) \ (-1, -1) \longrightarrow (\mathbf{1}, \mathbf{1}), & (\text{R4}) \ (1, -1) \longrightarrow (2, \mathbf{0}), & (\text{R7}) \ (-1, 1) \longrightarrow (3, 3), \\
(\text{R2}) \ (0, -1) \longrightarrow (\mathbf{1}, \mathbf{0}), & (\text{R5}) \ (1, 0) \longrightarrow (3, \mathbf{1}), & (\text{R8}) \ (0, 1) \longrightarrow (3, 2), \\
(\text{R3}) \ (0, 0) \longrightarrow (2, \mathbf{1}), & (\text{R6}) \ (-1, 0) \longrightarrow (2, 2), & (\text{R9}) \ (1, 1) \longrightarrow (4, 2).
\end{array}$$

Il est maintenant naturel de classifier ces neuf cas de rotation en les divisant en trois groupes, suivant si les nœuds x et y restent équilibrés dans T_1 .

- Les cas (R1) et (R2), où x et y restent équilibrés sont nommées *rotations conservatrice d'équilibre* ;
- Les cas (R3), (R4) et (R5), où y reste équilibré mais pas x sont nommées *rotations partiellement déséquilibrantes* ;
- Les (R6), (R7), (R8), et (R9) où x et y sont tous deux déséquilibrés sont nommées *rotations totalement déséquilibrantes*.

Ces observations mènent aux propriétés suivantes.

Proposition 8.1.2. *Soient T_0 et T_1 deux arbres binaires équilibrés. Alors,*

$$T_0 \prec T_1 \quad \text{implique} \quad h(T_0) = h(T_1). \quad (8.1.5)$$

Démonstration. Comme T_0 et T_1 sont tous deux équilibrés, la rotation modifie un sous-arbre S_0 de T_0 tel que la mesure de déséquilibre de la racine y de S_0 et celle du fils gauche x de y vérifient (R1) ou (R2). Soit S_1 l'arbre binaire obtenu par la rotation en y dans S_0 . Par un simple calcul des hauteurs de S_0 et de S_1 , on trouve $h(S_0) = h(S_1)$. Ainsi, comme une rotation modifie localement un arbre binaire, nous avons bien $h(T_0) = h(T_1)$. \square

Lemme 8.1.3. *Soit T_0 un arbre binaire équilibré et T_1 un arbre binaire déséquilibré tel que $T_0 \prec T_1$. Alors, il existe un nœud z dans T_1 tel que $d_{T_1}(z) \geq 2$ et les sous-arbres gauches et droits de z sont équilibrés.*

Démonstration. Soit y le nœud de T_0 tel que $T_0 \prec_y T_1$ et x son fils gauche. Si cette rotation est partiellement déséquilibrante, alors (R3), (R4) ou (R5) est vérifié et le nœud $z := x$ de T_1 répond au lemme. Si cette rotation est totalement déséquilibrante, alors (R6), (R7), (R8) ou (R9) est vérifié, et le nœud $z := y$ de T_1 répond au lemme. \square

Lemme 8.1.4. *Soit T_0 un arbre binaire et y un de ses nœuds tel que tous les sous-arbres de T_0 à droite de y sont équilibrés. Alors, pour tout arbre binaire T_1 tel que $T_0 \prec_y T_1$, tout sous-arbre de T_1 à droite de y est équilibré.*

Démonstration. Étant donné que l'opération de rotation ne modifie pas l'ordre infixe des nœuds et par définition de la relation \rightsquigarrow , si un sous-arbre S de T_1 est à droite de y , alors S est également à droite de y dans T_0 . Par hypothèse, S est équilibré dans T_0 . Il le reste donc dans T_1 . \square

8.1.3 Un invariant de déséquilibre

Témoins de déséquilibre

Rappelons qu'un sous-arbre S d'un arbre binaire T est à droite d'un nœud x de T si x n'est pas un nœud de S et l'intégralité des nœuds de S apparaissent après x dans le parcours infixe de T .

Soient T un arbre binaire, x un nœud de T et y le nœud le plus à gauche du sous-arbre de racine x dans T . Nous qualifions x de *témoin de déséquilibre* si les trois conditions suivantes sont vérifiées (voir figure 8.3) :

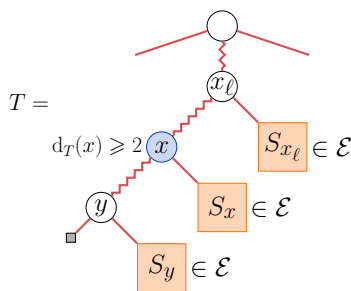


FIGURE 8.3 – Le nœud x est un témoin de déséquilibre de l'arbre binaire T .

(TD1) la mesure de déséquilibre de x est supérieure à 2 ;

(TD2) le sous-arbre gauche de x est équilibré;

(TD3) les sous-arbres à droite de y sont équilibrés.

Notons que (TD1) assure le fait qu'un arbre binaire qui possède un témoin de déséquilibre est déséquilibré.

Définissons maintenant une propriété supplémentaire de sorte que, lorsque les nœuds x et y la vérifient, tout arbre binaire T' tel que $T \leq_T T'$ possède encore un témoin de déséquilibre. De cette façon, en montrant que T' possède également cette propriété supplémentaire, nous montrerons du même coup qu'il est impossible de rééquilibrer T par une suite de rotations. Cette propriété, que nous qualifions de *propriété de conservation* est la suivante :

(PC) le mot des hauteurs du nœud y est admissible.

Nous dirons en outre que T possède un *invariant de déséquilibre* si T possède un témoin de déséquilibre satisfaisant la propriété de conservation. Les concepts nécessaires à la compréhension de cette propriété vont être présentés dans la suite du texte.

Mots des hauteurs

Soit T un arbre binaire, x_1 un de ses nœuds, (x_1, \dots, x_ℓ) la suite des ancêtres de x_1 qui sont à droite de x_1 et ordonnés de bas en haut, et $(S_{x_1}, \dots, S_{x_\ell})$ la suite des sous-arbres droits des x_i (voir la figure 8.4).

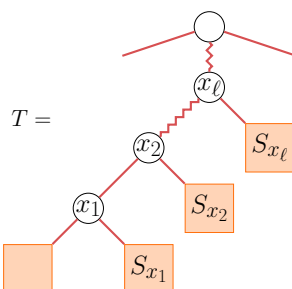


FIGURE 8.4 – La suite $(S_{x_1}, \dots, S_{x_\ell})$ des sous-arbres associée au nœud x_1 dans l'arbre binaire T .

Rappelons que la notation \mathbb{N}^* désigne l'ensemble des suites finies d'entiers naturels. Le mot $u_1 \dots u_\ell$ de \mathbb{N}^* défini par $u_i := h(S_{x_i})$ pour tout $i \in [\ell]$ est le *mot des hauteurs* de x_1 et est noté $\text{mh}_T(x_1)$. Il est pratique de poser $\text{mh}_T(x) := \epsilon$ lorsque x n'est pas un nœud de T . La figure 8.5 montre des exemples de mots des hauteurs associés à divers nœuds d'un arbre binaire.

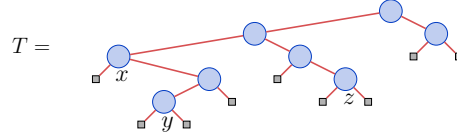


FIGURE 8.5 – Quelques exemples de mots des hauteurs dans l'arbre binaire T : $\text{mh}_T(x) = 221$, $\text{mh}_T(y) = 0021$ et $\text{mh}_T(z) = 01$.

Mots admissibles

Soit $\Theta : \mathbb{N}^2 \rightarrow \mathbb{N}$ la règle de réécriture définie par

$$\Theta(\mathbf{a} \cdot \mathbf{b}) := \begin{cases} \max\{\mathbf{a}, \mathbf{b}\} + 1 & \text{si } \mathbf{b} - \mathbf{a} \in \{-1, 0, 1\}, \\ \max\{\mathbf{a}, \mathbf{b}\} & \text{sinon.} \end{cases} \quad (8.1.6)$$

Cette règle de réécriture est étendue aux mots de \mathbb{N}^* par $\Theta(u) := \Theta(u_1 \cdot u_2) \cdot u_3 \dots u_{|u|}$. Si $0 \leq i \leq |u| - 1$, nous notons $\Theta^i(u)$ l'application itérée de Θ définie par

$$\Theta^i(u) := \begin{cases} u & \text{si } i = 0, \\ \Theta(\Theta^{i-1}(u)) & \text{sinon.} \end{cases} \quad (8.1.7)$$

Définition 8.1.5. Un mot $u \in \mathbb{N}^*$ est admissible si $|u| \leq 1$ ou bien tout mot v de l'ensemble

$$\{\Theta^i(u) : 0 \leq i \leq |u| - 2\} \quad (8.1.8)$$

vérifie $v_1 - 1 \leq v_2$. L'ensemble des mots admissibles est noté \mathcal{A} .

Pour tester si un mot u est admissible, il suffit de calculer itérativement les éléments de l'ensemble (8.1.8) suivant (8.1.7), et vérifier que chacun d'eux satisfait l'inégalité de la définition 8.1.5. Par exemple, en notant $\xrightarrow{\Theta}$ la règle de réécriture Θ , on s'aperçoit que $u := 00122$ est admissible. En effet,

$$00122 \xrightarrow{\Theta} 1122 \xrightarrow{\Theta} 222 \xrightarrow{\Theta} 32, \quad (8.1.9)$$

et $u_1 - 1 \leq u_2$ est respecté à chaque étape. Le mot 1234488 est également admissible :

$$01233778 \xrightarrow{\Theta} 2233778 \xrightarrow{\Theta} 333778 \xrightarrow{\Theta} 43778 \xrightarrow{\Theta} 5778 \xrightarrow{\Theta} 778 \xrightarrow{\Theta} 88. \quad (8.1.10)$$

D'un autre côté, 3444 n'est pas admissible puisque l'on a

$$3444 \xrightarrow{\Theta} 544 \xrightarrow{\Theta} 64, \quad (8.1.11)$$

et $6 - 1 \not\leq 4$.

Si u est un mot non vide, $\Omega(u)$ désigne la hauteur de u , qui est l'entier $\Theta^{|u|-1}(u)$. Nous avons ainsi par exemple $\Omega(00122) = 4$, $\Omega(01233778) = 9$ et $\Omega(3444) = 6$. Notons que nous pouvons déduire de la définition 8.1.5 qu'un mot $u \in \mathbb{N}^*$ de longueur supérieure à 2 est admissible si et seulement si pour toute écriture $u = v \cdot \mathbf{a} \cdot w$ où $v \in \mathbb{N}^+$, $\mathbf{a} \in \mathbb{N}$ et $w \in \mathbb{N}^*$, on a $\Omega(v) - 1 \leq \mathbf{a}$.

Propriétés des mots admissibles

Établissons trois lemmes sur les mots admissibles qui vont être utilisés pour établir le théorème 8.1.13, résultat principal de ce paragraphe.

Lemme 8.1.6. Si u est un mot admissible, alors, pour tout $i \in [|u| - 1]$, $u_i - 1 \leq u_{i+1}$.

Démonstration. Supposons que u est de la forme $u = v \cdot u_i \cdot u_{i+1} \cdot w$ où $v, w \in \mathbb{N}^*$ et $u_i - 1 > u_{i+1}$. Comme Θ transforme tout mot $\mathbf{a} \cdot \mathbf{b} \in \mathbb{N}^2$ en une lettre $\mathbf{c} \in \mathbb{N}$ de valeur supérieure à \mathbf{a} et \mathbf{b} , nous avons $\Omega(v \cdot u_i) \geq u_i$. Ceci implique $\Omega(v \cdot u_i) - 1 > u_{i+1}$, et ainsi $u \notin \mathcal{A}$, ce qui contredit l'hypothèse. \square

Lemme 8.1.7. *Tout préfixe et suffixe d'un mot admissible est admissible.*

Démonstration. Il est immédiat, par définition, que les préfixes d'un mot admissibles sont également admissibles.

Soient $u \in \mathcal{A}$ tel que $|u| \geq 2$ et w un suffixe non vide de u tel que $w \notin \mathcal{A}$. Par conséquent, w est de la forme $w = x \cdot \mathbf{a} \cdot y$ où $x \in \mathbb{N}^+$, $\mathbf{a} \in \mathbb{N}$, $y \in \mathbb{N}^*$ et $\Omega(x) - 1 > \mathbf{a}$. Le mot u est de la forme $u = v \cdot x \cdot \mathbf{a} \cdot y$ où $v \in \mathbb{N}^*$. Comme Θ transforme tout mot $\mathbf{a} \cdot \mathbf{b} \in \mathbb{N}^2$ en une lettre $\mathbf{c} \in \mathbb{N}$ de valeur supérieure à \mathbf{a} et \mathbf{b} , nous avons $\Omega(v \cdot x) \geq \Omega(x)$. Ceci implique $\Omega(v \cdot x) - 1 > \mathbf{a}$ et montre que $u \notin \mathcal{A}$, ce qui contredit l'hypothèse. \square

Lemme 8.1.8. *Si $u \cdot v$ est un mot admissible tel que $|v| \geq 2$, le mot $u \cdot \Theta(v)$ est également admissible.*

Démonstration. Si u est vide, le lemme est immédiatement vrai. Supposons donc que u est non vide. Le mot $u \cdot v$ est de la forme $u \cdot v = u \cdot \mathbf{a} \cdot \mathbf{b} \cdot w$ où $\mathbf{a}, \mathbf{b} \in \mathbb{N}$ et $w \in \mathbb{N}^*$. Posons $\mathbf{c} := \Theta(\mathbf{a} \cdot \mathbf{b}) = \Omega(\mathbf{a} \cdot \mathbf{b})$. Le mot $u \cdot \mathbf{c} \cdot w = u \cdot \Theta(v)$ est admissible dans le cas où les deux inégalités

$$\Omega(u) - 1 \leq \mathbf{c}, \quad (8.1.12)$$

et

$$\Omega(u \cdot \mathbf{c}) \leq \Omega(u \cdot \mathbf{a} \cdot \mathbf{b}), \quad (8.1.13)$$

sont établies. Comme $u \cdot \mathbf{a} \cdot \mathbf{b} \cdot w \in \mathcal{A}$, nous avons $\Omega(u) - 1 \leq \mathbf{a}$, et comme $\mathbf{c} = \Theta(\mathbf{a} \cdot \mathbf{b})$, nous avons $\mathbf{c} \geq \mathbf{a}$ et donc $\Omega(u) - 1 \leq \mathbf{c}$, ce qui montre (8.1.12). Posons à présent $\mathbf{d} := \Omega(u)$. Montrer (8.1.13) revient à montrer que $\Omega(\mathbf{d} \cdot \mathbf{c}) \leq \Omega(\mathbf{d} \cdot \mathbf{a} \cdot \mathbf{b})$, ce qui est équivalent à $\Omega(\mathbf{d} \cdot \Omega(\mathbf{a} \cdot \mathbf{b})) \leq \Omega(\mathbf{d} \cdot \mathbf{a} \cdot \mathbf{b})$. Cette relation est vraie dans le cas général pour tous $\mathbf{a}, \mathbf{b}, \mathbf{d} \in \mathbb{N}$, ce qui implique $u \cdot \Theta(v) \in \mathcal{A}$. \square

Mots des hauteurs et mots admissibles

Montrons à présent deux lemmes qui mettent en évidence le lien entre les mots des hauteurs des nœuds d'un arbre binaire et les mots admissibles.

Lemme 8.1.9. *Soient T un arbre binaire équilibré, x un de ses nœuds et u le mot des hauteurs de x . Alors, u est admissible et $\Omega(u) \leq h(T)$.*

Démonstration. Nous procédons par induction structurelle sur l'ensemble des arbres binaires équilibrés. Le lemme est vrai pour l'unique élément T de \mathcal{E}_1 puisque, en notant x son unique nœud, nous avons $u = \text{mh}_T(x) = 0$, qui est un mot admissible et vérifie $0 = \Omega(u) \leq h(T) = 1$.

Supposons que T est de la forme $T = L \wedge R$. Si x est un nœud de R , nous avons $u = \text{mh}_T(x) = \text{mh}_R(x)$, et par hypothèse d'induction, $u \in \mathcal{A}$ et $\Omega(u) \leq h(R)$. Comme $h(R) < h(T)$, le lemme est vérifié.

Si x est un nœud de L , nous avons $u = \text{mh}_T(x) = \text{mh}_L(x) \cdot h(R)$. Comme T est équilibré, $h(R) - h(L) \in \{-1, 0, 1\}$, et par hypothèse d'induction, $\Omega(\text{mh}_L(x)) \leq h(L)$. Par conséquent, $\Omega(\text{mh}_L(x)) - 1 \leq h(R)$. De plus, et encore par hypothèse d'induction, $\text{mh}_L(x) \in \mathcal{A}$, et ainsi $u \in \mathcal{A}$. Finalement, comme $\Omega(u) \leq h(R) + 1 \leq h(T)$, le lemme est vérifié. \square

Lemme 8.1.10. *Soient T un arbre binaire et y un de ses nœuds tel que $\text{mh}_T(y)$ est admissible et tous les sous-arbres de la suite $(S_{y_1}, \dots, S_{y_\ell})$ sont équilibrés. Alors, pour tout nœud x à droite de y dans T , le mot $\text{mh}_T(x)$ est admissible.*

Démonstration. Si x est un ancêtre de y , puisqu'il est à droite de y , y est dans le sous-arbre gauche de x . Par conséquent, $\text{mh}_T(x)$ est un suffixe de $\text{mh}_T(y)$, et d'après le lemme 8.1.7, $\text{mh}_T(x) \in \mathcal{A}$.

Autrement, il existe un indice $i \in [\ell]$ tel que x est un nœud de $S_{y_i} =: S$. Le mot des hauteurs de y est par définition de la forme $\text{mh}_T(y) = u \cdot h(S) \cdot v$ où $u, v \in \mathbb{N}^*$. Par le lemme 8.1.9, comme S est par hypothèse équilibré, $\text{mh}_S(x) \in \mathcal{A}$. D'après le lemme 8.1.7, $h(S) \cdot v \in \mathcal{A}$, et puisque d'après le lemme 8.1.9, $\Omega(\text{mh}_S(x)) \leq h(S)$, le mot $\text{mh}_T(x) = \text{mh}_S(x) \cdot v$ est également admissible. \square

Validité de l'invariant de déséquilibre

Nous donnons et montrons ici le résultat principal de ce paragraphe. Nous nous appuyons sur les deux propositions suivantes pour montrer que l'invariant de déséquilibre que nous avons défini est valide. Ceci nous permettra ainsi de montrer qu'il est impossible de rééquilibrer par l'intermédiaire de rotations un arbre binaire obtenu en appliquant une rotation déséquilibrante dans un arbre binaire équilibré.

Proposition 8.1.11. *Soient T_0 un arbre binaire équilibré et T_1 un arbre binaire déséquilibré obtenu en appliquant une rotation déséquilibrante dans T_0 . Alors, il existe un nœud x de T_1 qui est un témoin de déséquilibre et vérifie la propriété de conservation.*

Démonstration. Soient r le nœud de T_0 tel que $T_0 \prec_r T_1$, q le fils gauche de r dans T_0 , $S_0 := (A \wedge B) \wedge C$ le sous-arbre de T_0 de racine r et $S_1 := A \wedge (B \wedge C)$ le sous-arbre de T_1 de racine q (voir la figure 8.6). Nous allons exhiber dans cette démonstration le nœud x requis. D'après

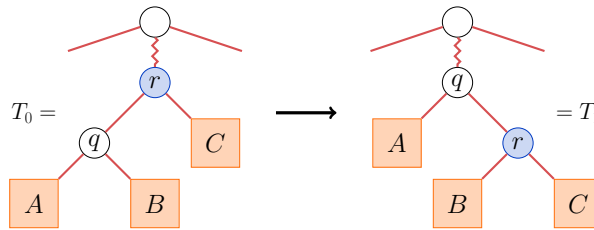


FIGURE 8.6 – Une rotation déséquilibrante en r est appliquée à l'arbre binaire équilibré T_0 .

le lemme 8.1.3, q ou r est déséquilibré dans T_1 et possède une mesure de déséquilibre positive. Nous avons par conséquent deux cas à examiner, suivant le type de la rotation déséquilibrante qui transforme T_0 en T_1 .

Cas 1. S'il s'agit d'une rotation partiellement déséquilibrante, posons $x := q$ et y le nœud le plus à gauche du sous-arbre de racine q dans T_1 . Puisque $d_{T_1}(x) \geq 2$, (TD1) est vérifié. De plus, comme T_0 est équilibré, par le lemme 8.1.4, les sous-arbres à droite de r sont équilibrés dans T_1 , et comme A et $B \wedge C$ sont équilibrés, (TD2) et (TD3) sont établis. Finalement, comme T_0 est équilibré, le lemme 8.1.9 montre que $\text{mh}_{T_0}(y) \in \mathcal{A}$. Nous avons

$$\text{mh}_{T_0}(y) = \text{mh}_A(y) \cdot h(B) \cdot h(C) \cdot v, \quad (8.1.14)$$

où $v \in \mathbb{N}^*$. De plus, nous avons

$$\text{mh}_{T_1}(y) = \text{mh}_A(y) \cdot h(B \wedge C) \cdot v = \text{mh}_A(y) \cdot \Theta(h(B) \cdot h(C)) \cdot v, \quad (8.1.15)$$

puisque $B \wedge C$ est équilibré. Ainsi, d'après le lemme 8.1.8, (PC) est satisfait.

Cas 2. S'il s'agit d'une rotation totalement déséquilibrante, posons $x := r$ et y le nœud le plus à gauche du sous-arbre de racine r dans T_1 . Puisque $d_{T_1}(x) \geq 2$, (TD1) est vérifié. De plus, comme T_0 est équilibré, par le lemme 8.1.4, les sous-arbres à droite de r sont équilibrés dans T_1 , et comme B est équilibré, (TD2) et (TD3) sont établis. Finalement, comme T_0 est équilibré, le lemme 8.1.9 montre que $\text{mh}_{T_0}(y) \in \mathcal{A}$. Nous avons

$$\text{mh}_{T_0}(y) = \text{mh}_B(y) \cdot h(C) \cdot v, \quad (8.1.16)$$

où $v \in \mathbb{N}^*$. De plus, nous avons

$$\text{mh}_{T_1}(y) = \text{mh}_B(y) \cdot h(C) \cdot v, \quad (8.1.17)$$

et ainsi, $\text{mh}_{T_1}(y) = \text{mh}_{T_0}(y)$, de sorte que (PC) est satisfait.

Il existe donc, dans tous les cas, un nœud x dans T_1 qui est un témoin de déséquilibre et qui satisfait la propriété de conservation. \square

Proposition 8.1.12. Soient T_1 un arbre binaire et x un nœud de T_1 qui est un témoin de déséquilibre. Si x satisfait la propriété de conservation, alors, pour tout arbre binaire T_2 obtenu en appliquant une rotation dans T_1 , il existe un nœud x' de T_2 qui est un témoin de déséquilibre et satisfait la propriété de conservation.

Démonstration. Soient y le nœud le plus à gauche du sous-arbre de racine x dans T_1 , r le nœud de T_1 tel que $T_1 \prec_r T_2$ et q le fils gauche de r dans T_1 . Nous allons exhiber, pour toute position de r par rapport à y dans T_1 , le nœud x' de T_2 requis. Si nécessaire, nous allons aussi exhiber le nœud y' de T_2 qui est le nœud le plus à gauche du sous-arbre de racine x' .

Cas 1. Si r est à gauche de y , la rotation en r ne modifie aucun des sous-arbres à droite de y . Donc, $x' := x$ est un témoin de déséquilibre qui satisfait la propriété de conservation dans T_2 .

Cas 2. Si r et q sont tous les deux ancêtres de y dans T_1 , posons C le sous-arbre droit de r et B le sous-arbre droit de q dans T_1 . Dans ce cas, T_2 est obtenu à partir de T_1 en remplaçant les sous-arbres B et C par $B \wedge C$ comme le montre la figure 8.7. Nous avons à présent trois

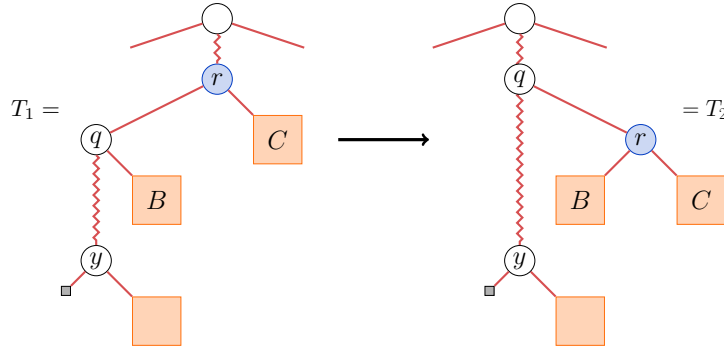


FIGURE 8.7 – Le deuxième cas : r est un ancêtre y et r est à droite de y .

possibilités suivant si $B \wedge C$ est équilibré et r est un ancêtre de x dans T_1 .

Cas 2.1. Si $B \wedge C$ est déséquilibré, on pose $x' := r$ et y' le nœud le plus à gauche de $B \wedge C$. Nous avons

$$\text{mh}_{T_1}(y) = u \cdot h(B) \cdot h(C) \cdot v, \quad (8.1.18)$$

où $u, v \in \mathbb{N}^*$. Comme x satisfait la propriété de conservation dans T_1 , $\text{mh}_{T_1}(y) \in \mathcal{A}$. Ainsi, d'après le lemme 8.1.6, nous avons $h(B) - 1 \leq h(C)$ de sorte que $d_{T_2}(x') \geq 2$ et (TD1) est

Cas 3.2. Si S_2 n'est pas équilibré, par la proposition 8.1.11, il existe un nœud x' de S_2 qui est un témoin de déséquilibre et satisfaisant la propriété de conservation, localement dans S_2 . Par conséquent, x' vérifie (TD1) et (TD2) dans T_2 . Il vérifie aussi (TD3) dans T_2 puisque par le lemme 8.1.4, les sous-arbres à droite de r restent équilibrés dans T_2 . Il reste à montrer que x' satisfait la propriété de conservation dans T_2 . On pose y' le nœud le plus à gauche du sous-arbre de racine x' dans T_2 . D'après la proposition 8.1.11, $w := \text{mh}_{S_2}(y') \in \mathcal{A}$, et par le lemme 8.1.9, w vérifie $\Omega(w) \leq h(S_1)$. Par hypothèse, $\text{mh}_{T_1}(y) \in \mathcal{A}$ et nous avons

$$\text{mh}_{T_1}(y) = u \cdot h(S_1) \cdot v, \quad (8.1.21)$$

où $u, v \in \mathbb{N}^*$. De plus, comme

$$\text{mh}_{T_2}(y') = w \cdot v, \quad (8.1.22)$$

nous avons $\text{mh}_{T_2}(y') \in \mathcal{A}$, ce qui établit (PC).

Il existe donc, dans tous les cas, un nœud x' dans T_2 qui est un témoin de déséquilibre et qui satisfait la propriété de conservation. \square

Théorème 8.1.13. *Soient T et T' deux arbres binaires équilibrés tels que $T \leq_T T'$. Alors, l'intervalle $[T, T']$ contient uniquement des arbres binaires équilibrés. En d'autres termes, tous les éléments supérieurs à un arbre binaire obtenu par une rotation déséquilibrante à partir d'un arbre binaire équilibré sont déséquilibrés.*

Démonstration. Soit T_0 un arbre binaire équilibré et T_1 un arbre binaire déséquilibré tel que $T_0 < T_1$. Par la proposition 8.1.11, il existe dans T_1 un témoin de déséquilibre x qui satisfait la propriété de conservation. La proposition 8.1.12 montre que tout arbre binaire T_2 tel que $T_1 \leq_T T_2$ est déséquilibré puisqu'il existe un nœud x' dans T_2 qui est un témoin de déséquilibre et qui satisfait la propriété de conservation. Par conséquent, la notion d'invariant de déséquilibre est valide, ce qui implique que l'ensemble des arbres binaires équilibrés est clos par intervalle dans le treillis de Tamari. \square

8.2 Motifs de déséquilibre et arbres binaires équilibrés

Dans cette partie, nous proposons un raffinement de l'ensemble des arbres binaires équilibrés en fonction de leur position dans le treillis de Tamari. Dans cette optique, nous définissons trois types d'arbres binaires équilibrés : les arbres binaires *maximaux*, *intérieurs* et *mixtes*. Il s'avère que les éléments de ces familles peuvent se décrire en termes de motifs d'arbres binaires évités ou forcés.

La notion de *motif*, pour décrire des ensembles de permutations (voir par exemple [BS00], ainsi que les références données en [Wil02]) ou de langages de mots (voir par exemple [Bur98]) est fondamentale. En effet, cette notion couplée avec un concept d'*évitement*, permet de décrire des ensembles d'objets. Les objets d'un tel ensemble sont simplement ceux qui ont la propriété commune d'éviter simultanément une collection de motifs donnés.

Plusieurs notions différentes de motifs d'arbre binaire et d'évitement ont été définies récemment — par exemple en [Lod05] et [Row10]. Nous proposons dans cette partie une nouvelle définition de motif d'arbre binaire qui se base sur les mesures de déséquilibre. Nous montrons que les arbres binaires équilibrés maximaux, intérieurs et mixtes admettent une caractérisation en termes de nos motifs d'arbre binaire. Ceci fait, nous construisons une grammaire synchrone (voir le chapitre 7) qui engendre les arbres binaires équilibrés maximaux et nous permet d'obtenir une équation fonctionnelle de point fixe pour les dénombrer.

8.2.1 Occurrences et motifs de déséquilibre

Définition 8.2.1. Un motif de déséquilibre est un arbre binaire non vide sans feuilles et étiqueté sur \mathbb{Z} .

Soit T un arbre binaire. On note T^d l'arbre binaire étiqueté de forme T dont les nœuds sont étiquetés par leur mesure de déséquilibre. Nous disons que T admet une *occurrence* d'un motif de déséquilibre p si une composante connexe de T^d est de même forme et possède les mêmes étiquettes que p . Par exemple, pour l'arbre binaire

$$T := \begin{array}{c} \text{[Diagram of a binary tree with 5 nodes and 4 edges, labeled } T \text{]} \end{array}, \quad (8.2.1)$$

nous avons

$$T^d = \begin{array}{c} \text{[Diagram of the same tree structure as } T \text{, but with nodes labeled with their imbalance measures: 2, 0, 0, 0, 0]} \end{array}, \quad (8.2.2)$$

ce qui montre que T admet au moins une occurrence des motifs de déséquilibre suivants :

$$\textcircled{0}, \textcircled{2}, \textcircled{0} \textcircled{0}, \textcircled{0} \textcircled{0}, \textcircled{2} \textcircled{0}, \textcircled{0} \textcircled{0} \textcircled{0}, \textcircled{2} \textcircled{0} \textcircled{0}, \textcircled{2} \textcircled{0} \textcircled{0}, \textcircled{2} \textcircled{0} \textcircled{0} \textcircled{0}. \quad (8.2.3)$$

Maintenant, étant donné un ensemble P de motifs de déséquilibre, nous pouvons nous intéresser à l'ensemble des arbres binaires qui *évitent* P , *i.e.*, les arbres binaires qui n'admettent aucune occurrence d'éléments de P . Par exemple, l'ensemble

$$\left\{ \textcircled{i} : i \notin \{-1, 0, 1\} \right\} \quad (8.2.4)$$

décrit l'ensemble des arbres binaires équilibrés, l'ensemble

$$\left\{ \textcircled{i} : i \neq 0 \right\} \quad (8.2.5)$$

décrit l'ensemble des arbres binaires parfaits, et l'ensemble

$$\left\{ \begin{array}{c} \textcircled{j} \\ \textcircled{i} \end{array} : i, j \in \mathbb{Z} \right\} \quad (8.2.6)$$

décrit l'ensemble des peignes droits.

Il existe un lien entre les grammaires synchrones et les motifs de déséquilibre puisque, comme nous l'avons illustré dans le paragraphe 7.3.3 du chapitre 7, ces dernières permettent d'engendrer des arbres à bourgeons tout en contrôlant les mesures de déséquilibre de ses nœuds. Par conséquent, les grammaires synchrones permettent d'engendrer des arbres binaires qui évitent certains motifs de déséquilibre. Cette remarque est utilisée dans ce qui suit pour dénombrer les arbres équilibrés maximaux.

8.2.2 Arbres équilibrés maximaux, intérieurs et mixtes

Arbres binaires équilibrés maximaux

Commençons par décrire un ensemble d'arbres binaires équilibrés et sa contrepartie dont les éléments se trouvent, approximativement parlant, aux extrémités du treillis de Tamari restreint aux arbres binaires équilibrés.

Définition 8.2.2. Un arbre binaire équilibré T_0 (resp. T_1) est maximal (resp. minimal) si, pour tout arbre binaire T_1 (resp. T_0) tel que $T_0 \prec T_1$, T_1 (resp. T_0) est déséquilibré. L'ensemble des arbres binaires équilibrés maximaux (resp. minimaux) est noté \mathcal{E}^{\max} (resp. \mathcal{E}^{\min}).

D'après le théorème 8.1.13, si T_0 (resp. T_1) est un arbre binaire équilibré maximal (resp. minimal), il n'existe aucun arbre binaire équilibré T_1 (resp. T_0) tel que $T_0 \leq_T T_1$. Les arbres binaires maximaux (resp. minimaux) sont ainsi les éléments maximaux (resp. minimaux) du treillis de Tamari restreint aux arbres binaires équilibrés.

Proposition 8.2.3. Un arbre binaire équilibré T est maximal si et seulement si il évite l'ensemble de motifs de déséquilibre

$$P_{\max} := \left\{ \begin{array}{c} \textcircled{-1} \\ \textcircled{-1} \end{array}, \begin{array}{c} \textcircled{-1} \\ \textcircled{0} \end{array} \right\}. \quad (8.2.7)$$

De même, un arbre binaire équilibré T est minimal si et seulement si il évite l'ensemble de motifs de déséquilibre

$$P_{\min} := \left\{ \begin{array}{c} \textcircled{1} \\ \textcircled{1} \end{array}, \begin{array}{c} \textcircled{1} \\ \textcircled{0} \end{array} \right\}. \quad (8.2.8)$$

Démonstration. Supposons que T est maximal. Alors, pour tout arbre binaire T_1 tel que $T \prec T_1$, T_1 est déséquilibré. Il est donc impossible de réaliser une rotation conservatrice d'équilibre dans T et, en regardant les différentes rotations possibles dans un arbre équilibré étudiées dans le paragraphe 8.1.2, il apparaît que T évite P_{\max} .

Réciproquement, supposons que T est un arbre binaire équilibré qui évite P_{\max} . Alors, pour tout arbre binaire T_1 tel que $T \prec T_1$, T_1 est déséquilibré puisque pour tout nœud y ayant un fils gauche x dans T , les mesures de déséquilibre de x et de y satisfont l'un des sept cas (R3)–(R9) du paragraphe 8.1.2. Ainsi, toute rotation dans T est déséquilibrante, ce qui implique que T est maximal.

La seconde partie de la proposition se prouve de manière analogue, en considérant les rotations gauches plutôt que les rotations droites. \square

Proposition 8.2.4. La série génératrice dénombrant les arbres binaires équilibrés maximaux suivant le nombre de feuilles des arbres est $\mathcal{S}_{\max}(x, 0, 0)$ où

$$\mathcal{S}_{\max}(x, y, z) = x + \mathcal{S}_{\max}(x^2 + xy + yz, x, xy). \quad (8.2.9)$$

Démonstration. Pour obtenir cette équation fonctionnelle de point fixe, considérons la grammaire synchrone $S_{\max} := (\{x, y, z\}, \textcircled{x}, R)$ où R contient les règles de substitution

$$x \mapsto \begin{array}{c} \textcircled{0} \\ \textcircled{x} \text{ } \textcircled{x} \end{array} + \begin{array}{c} \textcircled{1} \\ \textcircled{y} \text{ } \textcircled{x} \end{array} + \begin{array}{c} \textcircled{-1} \\ \textcircled{z} \text{ } \textcircled{y} \end{array}, \quad (8.2.10)$$

$$y \mapsto \textcircled{x}, \quad (8.2.11)$$

$$z \mapsto \begin{array}{c} \textcircled{1} \\ \textcircled{y} \text{ } \textcircled{x} \end{array}. \quad (8.2.12)$$

Nous pouvons employer la même méthode que celle utilisée dans la démonstration de la proposition 7.3.1 pour montrer que les arbres à bourgeons engendrés par S_{\max} qui ne contiennent que des bourgeons de type x ont leurs nœuds étiquetés par leurs mesures de déséquilibre. Par

conséquent, en identifiant dans ces arbres les bourgeons de type x avec des feuilles, S_{\max} engendre des arbres binaires équilibrés maximaux. En effet, par la proposition 8.2.3, les arbres engendrés doivent éviter les deux motifs de P_{\max} . Pour ce faire, il est nécessaire de contrôler la croissance des bourgeons de type x lorsqu'ils sont substitués par l'arbre à bourgeons D dont la racine possède une mesure de déséquilibre de -1 . En effet, si le fils gauche de la racine de D est substitué par un nœud ayant -1 ou 0 comme mesure de déséquilibre, un des deux motifs n'est pas évité. L'idée est donc de forcer la mesure de déséquilibre de ce nœud à être à 1 , ce qui est précisément le rôle des bourgeons de type z . Le rôle des bourgeons de type y est d'induire un délai sur la croissance de la branche qui le contient de sorte à pouvoir créer des nœuds ayant -1 ou 1 comme mesure de déséquilibre. De plus, nous pouvons montrer par induction structurale sur l'ensemble des arbres binaires équilibrés maximaux que S_{\max} permet d'engendrer tout arbre maximal.

En posant $y \leq_B z \leq_B x$, S_{\max} vérifie les hypothèses du lemme 7.1.9 et est par conséquent localement finie. Cette grammaire synchrone est aussi non ambiguë car elle satisfait les hypothèses du lemme 7.1.11. Toutes les racines des arbres à bourgeons apparaissant dans les règles de substitution de R sont en effets deux à deux différentes à cause de leur étiquetage.

Finalement, comme S_{\max} est localement finie et non ambiguë, par la proposition 7.2.3, nous obtenons l'équation fonctionnelle de point fixe annoncée et la série génératrice est obtenue par la spécialisation $y := 0$ et $z := 0$ dans le but d'ignorer les arbres à bourgeons qui contiennent des bourgeons du type y ou z . \square

L'équation fonctionnelle de point fixe de la proposition 8.2.4 permet de calculer les premières valeurs du nombre d'arbres binaires équilibrés maximaux par taille :

$$1, 1, 1, 1, 2, 2, 2, 4, 6, 9, 11, 13, 22, 38, 60, 89, 128, 183, 256, 353, 512, 805, 1336, 2221, \\ 3594, 5665, 8774, 13433, 20359, 30550, 45437, 67086, 98491, 144492, 213876. \quad (8.2.13)$$

Arbres binaires équilibrés intérieurs

Décrivons maintenant un ensemble d'arbres binaires équilibrés et sa contrepartie dont les éléments sont, approximativement parlant, au cœur du treillis de Tamari restreint aux arbres binaires équilibrés.

Définition 8.2.5. *Un arbre binaire équilibré T_0 (resp. T_1) est intérieur droit (resp. intérieur gauche) si tout arbre binaire T_1 (resp. T_0) tel que $T_0 \prec T_1$ est équilibré. L'ensemble des arbres binaires équilibrés intérieurs droits (resp. intérieurs gauches) est noté $\mathcal{E}^{\text{intd}}$ (resp. $\mathcal{E}^{\text{intg}}$).*

Proposition 8.2.6. *Un arbre binaire équilibré T est intérieur droit si et seulement si il évite l'ensemble de motifs de déséquilibre*

$$P_{\text{intd}} := \left\{ \begin{array}{c} \textcircled{0} \\ \textcircled{-1} \end{array}, \begin{array}{c} \textcircled{1} \\ \textcircled{-1} \end{array}, \begin{array}{c} \textcircled{0} \\ \textcircled{0} \end{array}, \begin{array}{c} \textcircled{1} \\ \textcircled{0} \end{array}, \begin{array}{c} \textcircled{-1} \\ \textcircled{1} \end{array}, \begin{array}{c} \textcircled{0} \\ \textcircled{1} \end{array}, \begin{array}{c} \textcircled{1} \\ \textcircled{1} \end{array} \right\}. \quad (8.2.14)$$

De même, un arbre binaire équilibré T est intérieur gauche si et seulement si il évite l'ensemble de motifs de déséquilibre

$$P_{\text{intg}} := \left\{ \begin{array}{c} \textcircled{0} \\ \textcircled{1} \end{array}, \begin{array}{c} \textcircled{-1} \\ \textcircled{1} \end{array}, \begin{array}{c} \textcircled{0} \\ \textcircled{0} \end{array}, \begin{array}{c} \textcircled{-1} \\ \textcircled{0} \end{array}, \begin{array}{c} \textcircled{1} \\ \textcircled{-1} \end{array}, \begin{array}{c} \textcircled{0} \\ \textcircled{-1} \end{array}, \begin{array}{c} \textcircled{-1} \\ \textcircled{-1} \end{array} \right\}. \quad (8.2.15)$$

Démonstration. Supposons que T est intérieur droit. Alors, pour tout arbre binaire T_1 tel que $T \prec T_1$, T_1 est équilibré. Par conséquent, seules des rotations conservatrices d'équilibre peuvent être réalisées dans T et donc, pour tout nœud y ayant un fils gauche x dans T , les mesures de

déséquilibre de x et de y vérifient (R1) ou (R2) du paragraphe 8.1.2. Ainsi, T évite les sept motifs de P_{intd} .

Réciproquement, supposons que T est un arbre binaire équilibré évitant les motifs de P_{intd} . Ainsi, pour tout nœud y de T qui possède un fils gauche x , les mesures de déséquilibre de x et y vérifient (R1) ou (R2) du paragraphe 8.1.2. Ainsi, la rotation en y donne un arbre binaire équilibré, ce qui implique que T est intérieur droit.

La seconde partie de la proposition se prouve de manière analogue, en considérant les rotations gauches plutôt que les rotations droites. \square

Les arbres binaires équilibrés intérieurs droits peuvent se dénombrer facilement suivant leur hauteur. En effet,

Proposition 8.2.7. *Le nombre a_h d'arbres binaires équilibrés intérieurs droits de hauteur h est*

$$a_h = \begin{cases} 1 & \text{si } h \in \{0, 1, 3\}, \\ 2 & \text{si } h = 2, \\ a_{h-1}a_{h-2} & \text{sinon.} \end{cases} \quad (8.2.16)$$

Démonstration. La proposition est vraie pour $0 \leq h \leq 3$ comme le montre une énumération exhaustive des possibilités.

Observons en premier lieu que si $T := L \wedge R$ est un arbre binaire équilibré intérieur droit de hauteur $h \geq 3$, alors L et R sont aussi intérieurs droits et la mesure de déséquilibre de la racine de T est -1 . En effet, si L ou R n'est pas intérieur droit, d'après la proposition 8.2.6, L ou R admettrait une occurrence d'un motif de P_{rint} et par conséquent, T aussi. De plus, si la mesure de déséquilibre de T n'est pas -1 , comme T est équilibré et que $h(T) \geq 3$, son sous-arbre gauche L serait non vide et T admettrait une occurrence d'un motif de P_{rint} .

Montrons finalement que pour tout entier $h \geq 4$ et tous arbres intérieurs droits L et R tels que $h(L) = h-1$ et $h(R) = h-2$, l'arbre binaire $T := L \wedge R$ est intérieur droit. Comme $h(L) \geq 3$, suivant ce que nous venons de montrer, la mesure de déséquilibre de la racine x de L est -1 . La mesure de déséquilibre de la racine y de T est également -1 , et ainsi, x et y ne forment pas un motif de P_{rint} dans T . De plus, la racine de R et le y de T ne forment pas non plus un motif de P_{rint} . Par conséquent, T est intérieur. Ceci termine la démonstration de (8.2.16). \square

Les premières valeurs de $(a_h)_{h \geq 0}$ sont

$$1, 1, 2, 1, 2, 2, 4, 8, 32, 256, 8192, 2097152, 17179869184. \quad (8.2.17)$$

Si l'on ne considère pas les trois premières valeurs, cette dernière est la suite **A000301** de [Slo]. De plus, nous avons $a_h = 2^{f_{h-3}}$ pour tout $h \geq 3$, où f_i est le i^{e} nombre de Fibonacci, défini par $f_i := i$ si $i \in \{0, 1\}$ et $f_i := f_{i-1} + f_{i-2}$ sinon.

Rappelons que l'ensemble des *arbres binaires de Fibonacci* [Knu98] est formé des éléments de la suite $(T_i)_{i \geq 0}$ où $T_0 := T_1 := \perp$ et $T_{i+2} := T_{i+1} \wedge T_i$. Une simple induction structurale sur l'ensemble des arbres binaires de Fibonacci permet de montrer que ces arbres sont également des arbres binaires équilibrés intérieurs droits.

Arbres binaires équilibrés mixtes

Caractérisons finalement les arbres binaires équilibrés qui ne sont ni maximaux (resp. minimaux) et ni intérieurs droits (resp. gauches).

Définition 8.2.8. *Un arbre binaire équilibré T_0 est mixte droit (resp. mixte gauche) s'il existe un arbre binaire équilibré T_1 et un arbre binaire déséquilibré T'_1 tel que $T_0 \prec T_1$ et $T_0 \prec T'_1$ (resp. $T_1 \prec T_0$ et $T'_1 \prec T_0$). L'ensemble des arbres binaires équilibrés mixtes droits (resp. mixtes gauches) est noté $\mathcal{E}^{\text{mixd}}$ (resp. $\mathcal{E}^{\text{mixg}}$).*

Proposition 8.2.9. *Un arbre binaire équilibré T est mixte droit (resp. mixte gauche) si et seulement si il admet au moins une occurrence du motif de déséquilibre de P_{\max} (resp. P_{\min}) et au moins une occurrence du motif de déséquilibre de P_{intd} (resp. P_{intg}).*

Démonstration. Supposons que T est mixte droit. Par définition, il est possible de réaliser une rotation conservatrice d'équilibre dans T . Il existe ainsi deux nœuds x et y dans T qui satisfont (R1) ou (R2) du paragraphe 8.1.2 et forment une occurrence d'un motif de P_{\max} . De plus, encore par définition, il est possible de réaliser une rotation déséquilibrante dans T . Il existe ainsi deux nœuds x' et y' dans T qui vérifient un des sept cas (R3)–(R9) du paragraphe 8.1.2 et forment une occurrence d'un motif de P_{intd} .

Réciproquement, si T admet à la fois une occurrence des motifs de P_{\max} et de P_{intd} , en regardant les neuf cas de rotation dans un arbre binaire équilibré étudiés dans le paragraphe 8.1.2, nous remarquons qu'il est possible de réaliser à la fois une rotation conservatrice d'équilibre et une rotation déséquilibrante dans T . Par conséquent T est mixte droit.

La seconde partie de la proposition se prouve de manière analogue, en considérant les rotations gauches plutôt que les rotations droites. \square

Notons que pour tout $n \geq 3$, l'ensemble \mathcal{E}_n admet les décompositions en unions disjointes

$$\mathcal{E}_n = \mathcal{E}_n^{\max} \uplus \mathcal{E}_n^{\text{intd}} \uplus \mathcal{E}_n^{\text{mixd}} = \mathcal{E}_n^{\min} \uplus \mathcal{E}_n^{\text{intg}} \uplus \mathcal{E}_n^{\text{mixg}}. \quad (8.2.18)$$

En effet, par définition, \mathcal{E}_n^{\max} et $\mathcal{E}_n^{\text{mixd}}$ sont disjoints, ainsi que $\mathcal{E}_n^{\text{intd}}$ et $\mathcal{E}_n^{\text{mixd}}$. Si T est un élément de $\mathcal{E}_n^{\max} \cap \mathcal{E}_n^{\text{intd}}$, T est le plus grand élément de \mathbb{T}_n et il s'agit donc d'un peigne droit. Comme T est équilibré, il possède nécessairement moins de trois nœuds, ce qui implique $n \leq 3$. Il est finalement évident que tout arbre binaire équilibré est soit maximal, soit intérieur droit, soit mixte droit.

8.3 Forme des intervalles d'arbres binaires équilibrés

Dans le paragraphe 8.1, nous avons établi le fait que les intervalles du treillis de Tamari de la forme $[T, T']$ contiennent uniquement des arbres binaires équilibrés si T et T' le sont. Dans ce paragraphe, nous nous intéressons de plus près aux intervalles de ce type. Il s'avère que ces intervalles ont une forme particulière dans le treillis de Tamari puisqu'ils ont, comme il est possible de l'observer dans des petits cas, la forme d'hypercubes. Après avoir démontré cette observation, nous utilisons les outils élaborés dans le chapitre 7 pour construire deux grammaires synchrones dont l'une engendre des structures arborescentes qui se trouvent être en bijection avec les intervalles d'arbres binaires équilibrés et l'autre, en bijection avec les intervalles maximaux d'arbres binaires équilibrés. Nous obtenons par ce biais deux équations fonctionnelles de point fixe pour les séries génératrices du nombre d'intervalles pour l'une, et d'intervalles maximaux pour l'autre, d'arbres binaires équilibrés.

8.3.1 Intervalles d'arbres binaires équilibrés et hypercubes

Lemme 8.3.1. *Soient T_0 et T_1 deux arbres binaires équilibrés tels que $T_0 \leq_{\mathbb{T}} T_1$ et y un nœud de T_0 . Alors,*

- (i) *si la rotation en y dans T_0 est déséquilibrante, la rotation en y dans T_1 , si elle est possible, est également déséquilibrante ;*
- (ii) *si y n'a pas de fils gauche dans T_0 , y n'a pas de fils gauche dans T_1 .*

Démonstration. (i) : si la rotation en y dans T_0 est déséquilibrante, alors les mesures de déséquilibre de y et de son fils gauche x satisfont les cas (R1) ou (R2) du paragraphe 8.1.2. Par conséquent, pour modifier ces mesures de déséquilibre, il faut nécessairement modifier les hauteurs des sous-arbres de x et de y par des rotations. D'après la proposition 8.1.2, ces rotations

déséquilibreraient l'arbre binaire obtenu. De plus, par le théorème 8.1.13, il serait impossible de le rééquilibrer par l'intermédiaire de rotations. Ceci montre donc que si y possède un fils gauche dans T_1 , toute rotation en y est déséquilibrante.

(ii) : ce point est immédiatement impliqué par la définition de l'opération de rotation et par le fait que l'application d'une rotation dans un arbre binaire ne modifie pas l'ordre infixe sur ses nœuds. \square

Le lemme 8.3.1 montre donc que pour tous arbres binaires équilibrés T_0 et T_1 tels que $T_0 \leq_T T_1$, et tout nœud y de T_0 , une rotation en y ne peut pas être conservatrice d'équilibre dans T_1 si elle ne l'est pas déjà dans T_0 .

Lemme 8.3.2. *Soient T_0 et T_1 deux arbres binaires équilibrés et y un nœud de T_0 tel que $T_0 \prec_y T_1$. Alors, en notant x le fils gauche de y dans T_0 , pour tout arbre binaire équilibré T_2 tel que $T_1 \leq_T T_2$, toute rotation en x et en y dans T_2 est déséquilibrante.*

Démonstration. Puisque T_1 est obtenu par une rotation en y dans T_0 , nous avons deux cas à considérer suivant les mesures de déséquilibre de x et de y dans T_0 . Si $d_{T_0}(x) = d_{T_0}(y) = -1$, alors $d_{T_1}(x) = d_{T_1}(y) = 1$ et toute rotation en x ou en y est déséquilibrante dans T_1 . Ainsi, par le lemme 8.3.1, toute rotation en x ou en y est déséquilibrante dans T_2 . Si $d_{T_0}(x) = 0$ et $d_{T_0}(y) = 1$, alors $d_{T_1}(x) = 1$ et $d_{T_1}(y) = 0$. Pour la même raison que dans le cas précédent, toute rotation en x ou en y est déséquilibrante dans T_2 . \square

Rappelons qu'un *hypercube* de dimension k peut être vu comme un poset dont les éléments sont les sous-ensembles d'un ensemble $\{e_1, \dots, e_k\}$, et la relation d'ordre, l'inclusion ensembliste. On note \mathbb{H}_k l'hypercube de dimension k .

La forme des intervalles d'arbres binaires équilibrés est élucidée par le théorème suivant.

Théorème 8.3.3. *Soient T_0 et T_1 deux arbres binaires équilibrés tels que $T_0 \leq_T T_1$. Alors, le poset $([T_0, T_1], \leq_T)$ est isomorphe à l'hypercube \mathbb{H}_k , où k est le nombre de rotations requises pour obtenir T_1 à partir de T_0 .*

Démonstration. Remarquons en premier lieu que d'après le théorème 8.1.13, l'intervalle $I := [T_0, T_1]$ contient uniquement des arbres binaires équilibrés. Par conséquent, les relations de couverture de I correspondent à rotations conservatrices d'équilibre.

Soit R l'ensemble des nœuds y de T_0 tels qu'une rotation en y est nécessaire pour obtenir T_1 à partir de T_0 . Par le lemme 8.3.2, R est bien défini — il ne s'agit pas d'un multi-ensemble — et si $y \in R$ alors, en notant x le fils gauche de y dans T_0 , nous avons $x \notin R$. Ceci implique que T_1 peut être obtenu à partir de T_0 en réalisant une suite arbitraire de rotations en chacun des nœuds de R .

Définissons maintenant une bijection entre I et l'ensemble des sous-ensembles de R . Soit $T \in I$. Par définition, il est possible d'obtenir T à partir de T_0 en réalisant une certaine suite s_0 de rotations depuis T_0 . Soit R_0 l'ensemble des nœuds qui apparaissent dans s_0 . Il est de plus possible d'obtenir T_1 à partir de T en réalisant une certaine suite s_1 de rotations depuis T . Soit R_1 l'ensemble des nœuds qui apparaissent dans s_1 . Par le lemme 8.3.1, nous avons $R = R_0 \uplus R_1$, et ainsi $R_0 \subset R$. L'ensemble R_0 caractérise T . Inversement, pour tout sous-ensemble $R_0 \subseteq R$, nous pouvons construire un unique arbre binaire $T \in I$. En effet, T est obtenu en réalisant dans T_0 une suite arbitraire de rotations en chacun des nœuds de R_0 . Ceci est bien défini, par définition de R .

Nous venons ainsi de montrer que I est isomorphe à \mathbb{H}_k où k est le nombre de rotations nécessaires pour obtenir T_1 à partir de T_0 . \square

Les premiers sous-posets du treillis de Tamari restreint aux arbres binaires équilibrés sont représentés par la figure 8.9.

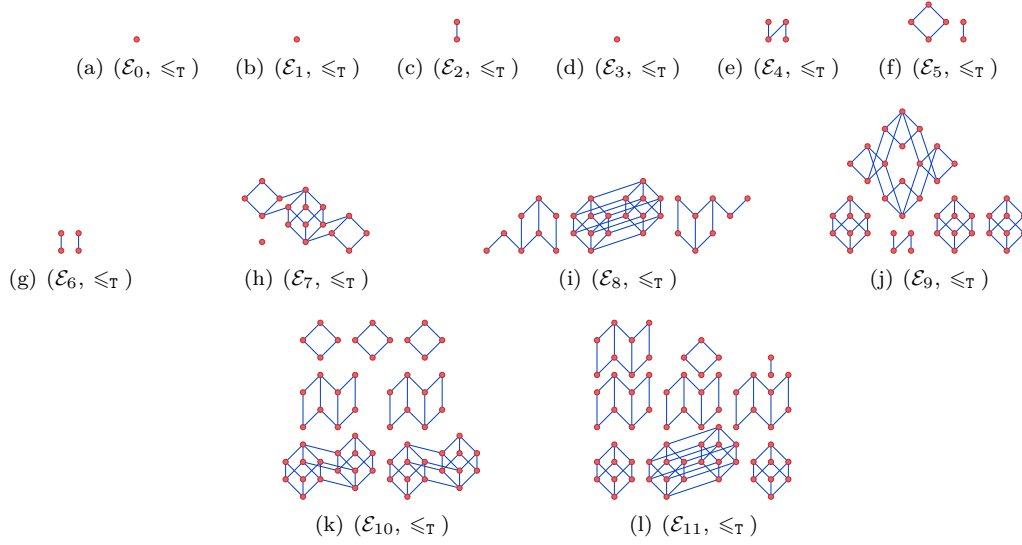


FIGURE 8.9 – Les diagrammes de Hasse des douze premiers sous-posets du treillis de Tamari restreint aux arbres binaires équilibrés.

8.3.2 Énumération des intervalles d'arbres binaires équilibrés

Proposition 8.3.4. *La série génératrice dénombrant les intervalles d'arbres binaires équilibrés dans le treillis de Tamari suivant le nombre de feuilles des arbres qui les constituent est $S_{ie}(x, 0, 0)$ où*

$$S_{ie}(x, y, z) = x + S_{ie}(x^2 + 2xy + yz, x, x^2 + xy). \quad (8.3.1)$$

Démonstration. Soit $I := [T_0, T_1]$ un intervalle d'arbres binaires équilibrés et R l'ensemble des nœuds y de T_0 tels qu'une rotation en y est nécessaire pour obtenir T_1 à partir de T_0 . La démonstration du théorème 8.3.3 montre que R est bien défini et aussi que I est entièrement déterminé par T_0 dans lequel les nœuds de R sont marqués.

Dans le but d'engendrer ces objets, nous considérons une grammaire synchrone qui engendre des arbres à bourgeons dont les nœuds sont de deux sortes : des nœuds non marqués, représentés par \bullet , et des nœuds marqués, représentés par \blacksquare . Soit la grammaire synchrone $S_{ie} := (\{x, y, z\}, \langle x \rangle, L)$ où L contient les règles de substitution

$$x \mapsto \begin{array}{c} (-1) \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \begin{array}{c} (0) \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \begin{array}{c} (1) \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \begin{array}{c} \boxed{-1} \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array}, \quad (8.3.2)$$

$$y \mapsto \langle x \rangle, \quad (8.3.3)$$

$$z \mapsto \begin{array}{c} (0) \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array} + \begin{array}{c} (-1) \\ \swarrow \quad \searrow \\ \bullet \quad \bullet \end{array}. \quad (8.3.4)$$

Nous pouvons employer la même méthode que celle utilisée dans la démonstration de la proposition 7.3.1 pour montrer que les arbres à bourgeons engendrés par S_{ie} qui ne contiennent que des bourgeons de type x ont leurs nœuds étiquetés par leurs mesures de déséquilibre. Par conséquent, en identifiant dans ces arbres les bourgeons de type x avec des feuilles, S_{ie} engendre des arbres binaires équilibrés tels que tout nœud r ayant -1 comme mesure de déséquilibre

peut être marqué à condition que son fils gauche ait -1 ou 0 comme mesure de déséquilibre — rappelons que de cette façon une rotation en r est conservatrice d'équilibre. En effet, si un bourgeon de type x est substitué par un nœud marqué, ce nœud marqué possède un bourgeon de type z en tant que fils gauche et tout bourgeon de type z ne peut être substitué que par un nœud non marqué de mesure de déséquilibre -1 ou 0 . Le rôle des bourgeons de type y est d'induire un délai sur la croissance de la branche qui le contient de sorte à pouvoir créer des nœuds ayant -1 ou 1 comme mesure de déséquilibre.

En posant $y \leq_B z \leq_B x$, S_{ie} vérifie les hypothèses du lemme 7.1.9 et est donc localement finie. Cette grammaire synchrone est également non ambiguë puisqu'elle satisfait les hypothèses du lemme 7.1.11. En effet, les racines des arbres à bourgeons qui apparaissent dans les membres droits des règles de substitution de L sont deux à deux différentes à cause de leur étiquetage et de leur marquage.

Finalement, comme S_{ie} est localement finie et non ambiguë, par la proposition 7.2.3, nous obtenons l'équation fonctionnelle de point fixe annoncée et la série génératrice est obtenue par la spécialisation $y := 0$ et $z := 0$ dans le but d'ignorer les arbres à bourgeons qui contiennent des bourgeons du type y ou z . \square



L'équation fonctionnelle de point fixe de la proposition 8.3.4 permet de calculer les premières valeurs du nombre d'intervalles d'arbres équilibrés dans le treillis de Tamari :

$$1, 1, 3, 1, 7, 12, 6, 52, 119, 137, 195, 231, 1019, 3503, 6593, 12616, 26178, 43500, \\ 64157, 94688, 232560, 817757, 2233757, 5179734, 11676838, 24867480. \quad (8.3.5)$$

Un intervalle $[T_0, T_1]$ est dit *maximal* si T_0 est un arbre binaire équilibré minimal et T_1 est un arbre binaire équilibré maximal.

Proposition 8.3.5. *La série génératrice dénombrant les intervalles maximaux d'arbres binaires équilibrés dans le treillis de Tamari suivant le nombre de feuilles des arbres qui les constituent est $\mathcal{S}_{iem}(x, 0, 0, 0)$ où*

$$\mathcal{S}_{iem}(x, y, z, t) = x + \mathcal{S}_{iem}(x^2 + 2yt + yz, x, x^2 + xy, yt + yz). \quad (8.3.6)$$

Démonstration. Soit $I := [T_0, T_1]$ un intervalle maximal d'arbres binaires équilibrés. Par le même argument que celui qui a été utilisé dans la démonstration de la proposition 8.3.4, il apparaît que I peut être encodé par l'arbre T_0 couplé à l'ensemble R des nœuds y de T_0 tels qu'une rotation en y est nécessaire pour obtenir T_1 à partir de T_0 . Nous allons donc encoder I par un arbre à bourgeons de la même forme que T_0 et qui est constitué de nœuds non marqués, représentés par , et de nœuds marqués, représentés par . Cette fois-ci, comme I est maximal, T_0 est minimal et l'arbre à bourgeons représentant I correspond à un arbre minimal. De plus, comme T_1 est maximal, il est impossible de réaliser une rotation conservatrice d'équilibre dans T_1 . Ceci implique qu'un nœud y de l'arbre à bourgeons qui encode I est marqué si et seulement si son fils gauche x n'est pas marqué et une rotation en y est conservatrice d'équilibre.

Nous utilisons la grammaire synchrone $S_{iem} := (\{x, y, z, u, v\}, \textcircled{x}, L)$ dans le but d'engendrer

ces objets, où L contient les règles de substitution

$$x \mapsto \begin{array}{c} (-1) \\ \swarrow \quad \searrow \\ v \quad y \end{array} + \begin{array}{c} (0) \\ \swarrow \quad \searrow \\ x \quad x \end{array} + \begin{array}{c} (1) \\ \swarrow \quad \searrow \\ y \quad u \end{array} + \begin{array}{c} \boxed{-1} \\ \swarrow \quad \searrow \\ z \quad y \end{array}, \quad (8.3.7)$$

$$y \mapsto \begin{array}{c} x \end{array}, \quad (8.3.8)$$

$$z \mapsto \begin{array}{c} (-1) \\ \swarrow \quad \searrow \\ x \quad y \end{array} + \begin{array}{c} (0) \\ \swarrow \quad \searrow \\ x \quad x \end{array}, \quad (8.3.9)$$

$$u \mapsto \begin{array}{c} (-1) \\ \swarrow \quad \searrow \\ v \quad y \end{array} + \begin{array}{c} \boxed{-1} \\ \swarrow \quad \searrow \\ z \quad y \end{array}, \quad (8.3.10)$$

$$v \mapsto \begin{array}{c} (1) \\ \swarrow \quad \searrow \\ y \quad u \end{array} + \begin{array}{c} \boxed{-1} \\ \swarrow \quad \searrow \\ z \quad y \end{array}. \quad (8.3.11)$$

Le même méthode que celle employée dans la démonstration de la proposition 7.3.1 permet de montrer que les arbres à bourgeons engendrés par S_{iem} qui ne contiennent que des bourgeons de type x ont leurs nœuds étiquetés par leurs mesures de déséquilibre. Par conséquent, en identifiant dans ces arbres les bourgeons de type x avec des feuilles, S_{iem} engendre des arbres binaires équilibrés minimaux qui sont marqués de manière maximale. En effet, d'après la proposition 8.2.3, les arbres engendrés doivent éviter les deux motifs de P_{\min} . Pour ce faire, il est nécessaire de contrôler la croissance des bourgeons de type x lorsqu'ils sont substitués par l'arbre à bourgeons D dont la racine possède une mesure de déséquilibre de 1. En effet, si le fils droit de la racine de D est substitué par un nœud ayant 0 ou 1 comme mesure de déséquilibre, un des deux motifs n'est pas évité. L'idée est de forcer la mesure de déséquilibre de ce nœud à être à -1 , qui est précisément le rôle des bourgeons de type u . De plus, si un bourgeon de type x est substitué par un nœud a non marqué et étiqueté par -1 , dans le but d'engendrer un arbre marqué de manière maximale, le fils gauche de a doit être marqué, ou bien étiqueté par 1 — dans ce cas, une rotation en a est déséquilibrante. C'est le rôle des bourgeons de type v . Les bourgeons de type z apparaissent dans les règles de substitution uniquement en tant que fils gauches d'un nœud marqué a et sont substitués exclusivement par des nœuds ayant -1 ou 0 comme mesure de déséquilibre, qui sont les seules autorisées puisque la rotation en a est conservatrice d'équilibre. Le rôle des bourgeons de type y est d'induire un délai sur la croissance de la branche qui le contient de sorte à pouvoir créer des nœuds ayant -1 ou 1 comme mesure de déséquilibre.

En posant $y \leq_B v \leq_B u \leq_B z \leq_B x$, S_{iem} satisfait les hypothèses du lemme 7.1.9, ce qui montre qu'elle est localement finie. Cette grammaire synchrone est de plus non ambiguë puisqu'elle vérifie les hypothèses du lemme 7.1.11. Les racines de chacun des arbres à bourgeons des règles de substitution de L sont en effet différentes à cause de leur étiquettes ou de leur marquage.

D'après la proposition 7.2.3, l'équation fonctionnelle de point fixe associée à S_{iem} est

$$F(x, y, z, u, v) = x + F(x^2 + yu + yv + yz, x, x^2 + xy, yv + yz, yu + yz), \quad (8.3.12)$$

et, comme les variables u et v jouent le même rôle, nous obtenons l'équation fonctionnelle annoncée. La série génératrice est finalement obtenue par la spécialisation $y := 0$, $z := 0$ et $t := 0$ dans le but d'ignorer les arbres à bourgeons qui contiennent des bourgeons du type y , z , u , ou v . \square

L'équation fonctionnelle de point fixe de la proposition 8.3.5 permet de calculer les premières

valeurs du nombre d'intervalles maximaux d'arbres équilibrés dans le treillis de Tamari :

$$\begin{aligned} &1, 1, 1, 1, 3, 2, 2, 6, 9, 15, 15, 17, 41, 77, 125, 178, 252, 376, 531, 740, 1192, 2179, 4273, \\ &7738, 13012, 20776, 32389, 49841, 75457, 113011, 168888, 252881, 379348. \end{aligned} \quad (8.3.13)$$

Un raffinement de S_{iem} permet de prendre en compte les dimensions des hypercubes isomorphes aux intervalles maximaux d'arbres équilibrés dénombrés. D'après le théorème 8.3.3, la dimension de l'hypercube isomorphe à un intervalle $I := [T, T']$ correspond au nombre de rotations nécessaires pour transformer T en T' . D'après la démonstration de la proposition 8.3.5, cette quantité correspond au nombre de nœuds marqués dans l'arbre à bourgeons qui encode I . Il suffit donc de tenir compte des règles de substitution qui engendrent un nœud marqué. Nous utilisons pour cela un paramètre ξ , et obtenons l'équation fonctionnelle de point fixe

$$S_{\text{iem}}(x, y, z, t, \xi) = x + S_{\text{iem}}(x^2 + 2yt + yz\xi, x, x^2 + xy, yt + yz\xi, \xi). \quad (8.3.14)$$

Les premiers coefficients de x^i dans $P := S_{\text{iem}}(x, 0, 0, 0, \xi)$ sont

$$[x^1] P = 1, \quad (8.3.15) \quad [x^8] P = 1 + 4\xi^2 + \xi^3, \quad (8.3.22)$$

$$[x^2] P = 1, \quad (8.3.16) \quad [x^9] P = 4\xi + 4\xi^2 + \xi^4, \quad (8.3.23)$$

$$[x^3] P = \xi, \quad (8.3.17) \quad [x^{10}] P = 3\xi + 9\xi^2 + 3\xi^3, \quad (8.3.24)$$

$$[x^4] P = 1, \quad (8.3.18) \quad [x^{11}] P = 9\xi^2 + 6\xi^3, \quad (8.3.25)$$

$$[x^5] P = 3\xi, \quad (8.3.19) \quad [x^{12}] P = \xi + 13\xi^2 + 2\xi^3 + \xi^4, \quad (8.3.26)$$

$$[x^6] P = \xi + \xi^2, \quad (8.3.20) \quad [x^{13}] P = 6\xi + 4\xi^2 + 16\xi^3 + 15\xi^4, \quad (8.3.27)$$

$$[x^7] P = 2\xi, \quad (8.3.21) \quad [x^{14}] P = 2\xi + 18\xi^2 + 31\xi^3 + 12\xi^4 + 14\xi^5, \quad (8.3.28)$$

Ainsi par exemple, le coefficient de x^{12} de $S_{\text{iem}}(x, 0, 0, 0, \xi)$ renseigne sur le fait que dans le poset $(\mathcal{E}_{11}, \leq_T)$, il y a exactement un hypercube maximal de dimension 1, treize hypercubes maximaux de dimension 2, deux hypercubes maximaux de dimension 3 et un hypercube maximal de dimension 4 (voir la figure 8.9).

Notons d'ailleurs que la proposition 8.1.2 implique que les arbres binaires d'une même composante connexe des posets (\mathcal{E}_n, \leq_T) sont de même hauteur. La réciproque est en revanche fausse : il existe deux composantes connexes dans (\mathcal{E}_5, \leq_T) et tous les éléments de \mathcal{E}_5 sont de même hauteur.

8.4 Autres familles d'arbres binaires closes par intervalle

Dans ce paragraphe, nous nous intéressons à d'autres ensembles d'arbres binaires clos par intervalle dans le treillis de Tamari. Nous commençons par considérer une généralisation des arbres binaires équilibrés, où les mesures de déséquilibre autorisées ne sont pas seulement -1 , 0 et 1 mais dans un ensemble fixé. Nous regardons ensuite les arbres binaires équilibrés en nombre de nœuds, puis les arbres binaires ayant une canopée fixée, pour terminer avec les arbres binaires ayant un indice de Narayana fixé.

8.4.1 Arbres binaires équilibrés généralisés

Définitions

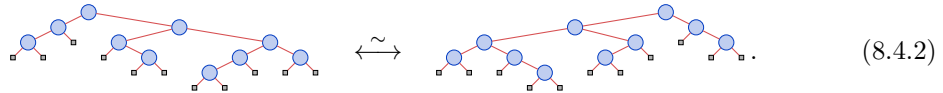
Soit V un sous-ensemble de \mathbb{Z} . Un arbre binaire T est V -équilibré si pour chacun de ses nœuds x , $d_T(x) \in V$. On note \mathcal{E}^V la classe combinatoire de ces arbres, où la *taille* est le nombre de nœuds. Remarquons que l'ensemble des arbres binaires équilibrés est $\mathcal{E}^{[-1,1]}$. Il est évident

que $\mathcal{E}^V \neq \emptyset$ implique $0 \in V$ puisqu'un arbre binaire possède obligatoirement au moins un nœud avec des sous-arbres gauches et droits vides. Dans le cas contraire \mathcal{E}^V serait vide. Une question naturelle à propos des arbres binaires V -équilibrés est de déterminer les ensembles V pour lesquels \mathcal{E}^V est clos par intervalle dans le treillis de Tamari.

Soit T un arbre binaire. L'application $\sim: \mathcal{T} \rightarrow \mathcal{T}$ est définie comme suit. L'arbre binaire T^\sim est obtenu en échangeant les sous-arbres gauches et droits de chacun des nœuds de T . Plus formellement,

$$T^\sim := \begin{cases} R^\sim \wedge L^\sim & \text{si } T = L \wedge R, \\ \perp & \text{sinon } (T = \perp). \end{cases} \quad (8.4.1)$$

par exemple,



Si V est un sous-ensemble de \mathbb{Z} , notons également par V^\sim l'ensemble $\{-v : v \in V\}$.

Une symétrie

Lemme 8.4.1. *Soient T_0 et T_1 deux arbres binaires tels que $T_0 \leq_T T_1$. Alors, $T_1^\sim \leq_T T_0^\sim$.*

Démonstration. Supposons $S_0 \prec S_1$ où $S_0 = (A \wedge B) \wedge C$ et $S_1 = A \wedge (B \wedge C)$. Par conséquent, $S_1^\sim = (C^\sim \wedge B^\sim) \wedge A^\sim$ et $S_0^\sim = C^\sim \wedge (B^\sim \wedge A^\sim)$. Ainsi, $S_1^\sim \prec S_0^\sim$, et le résultat suit du fait que \leq_T est la clôture réflexive et transitive de \prec (voir la définition 1.3.2 du chapitre 1). \square

Lemme 8.4.2. *Pour tout $V \subseteq \mathbb{Z}$, l'application \sim définit une bijection entre les ensembles \mathcal{E}^V et \mathcal{E}^{V^\sim} .*

Démonstration. Il est immédiat, d'après sa définition, que \sim est une involution. Il reste donc à montrer que si $T \in \mathcal{E}^V$, alors $T^\sim \in \mathcal{E}^{V^\sim}$. Soit x un nœud de T et L (resp. R) le sous-arbre gauche (resp. droit) de x . Nous avons $v := d_T(x) = h(R) - h(L) \in V$. Dans T^\sim , on a

$$d_{T^\sim}(x) = h(L^\sim) - h(R^\sim) = h(L) - h(R) = -v \in V^\sim. \quad (8.4.3)$$

Ainsi, $T^\sim \in \mathcal{E}^{V^\sim}$. \square

Proposition 8.4.3. *Pour tout $V \subseteq \mathbb{Z}$, l'ensemble \mathcal{E}^V est clos par intervalle dans le treillis de Tamari si et seulement si l'ensemble \mathcal{E}^{V^\sim} l'est également.*

Démonstration. Supposons que \mathcal{E}^{V^\sim} est clos par intervalle dans le treillis de Tamari. Par l'absurde, supposons qu'il existe $T_0, T_2 \in \mathcal{E}^V$ et $T_1 \notin \mathcal{E}^V$ tels que $T_0 \leq_T T_1 \leq_T T_2$. D'après le lemme 8.4.1, nous avons $T_2^\sim \leq_T T_1^\sim \leq_T T_0^\sim$, et d'après le lemme 8.4.2, $T_0^\sim, T_2^\sim \in \mathcal{E}^{V^\sim}$ et $T_1^\sim \notin \mathcal{E}^{V^\sim}$. Ceci implique que \mathcal{E}^{V^\sim} n'est pas clos par intervalle dans le treillis de Tamari, ce qui est en contradiction avec notre hypothèse. \square

Arbres binaires $\{0, 1\}$ -équilibrés

En utilisant les méthodes développées dans le chapitre 7, nous obtenons l'équation fonctionnelle de point fixe

$$\mathcal{S}_{01}(x, y) = x + \mathcal{S}_{01}(x^2 + xy, x), \quad (8.4.4)$$

qui permet de dénombrer les arbres binaires $\{0, 1\}$ -équilibrés suivant leur nombre de feuilles. La série génératrice de ces arbres est la spécialisation $\mathcal{S}_{01}(x, 0)$. Ses premiers coefficients sont

$$1, 1, 1, 1, 1, 2, 2, 2, 3, 5, 7, 9, 11, 13, 17, 26, 42, 66, 97, 134, 180, 241, 321, 424, 564, 774, 1111. \quad (8.4.5)$$

Proposition 8.4.4. *L'ensemble des arbres binaires $\{0, 1\}$ -équilibrés est clos par intervalle dans le treillis de Tamari.*

Démonstration. Soit $T_0 \in \mathcal{E}^{\{0,1\}}$. Comme T_0 est constitué uniquement de nœuds ayant 0 ou 1 comme mesure de déséquilibre, il n'est possible de réaliser dans T_0 que des rotations du type (R3), (R5), (R8), ou (R9) définies dans le paragraphe 8.1.2. Puisque ces rotations sont déséquilibrantes, tout arbre binaire T_1 tel que $T_0 \prec T_1$ est déséquilibré, et ainsi, $T_1 \notin \mathcal{E}^{\{0,1\}}$. Par le théorème 8.1.13, tout arbre binaire T_2 tel que $T_1 \leq_T T_2$ est déséquilibré, et donc à plus forte raison, $T_2 \notin \mathcal{E}^{\{0,1\}}$. Par conséquent, $\mathcal{E}^{\{0,1\}}$ est clos par intervalle dans le treillis de Tamari. \square

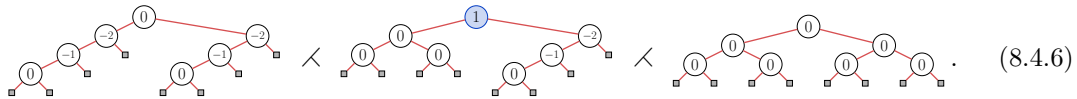
La démonstration de la proposition 8.4.4 montre également que que l'application d'une rotation dans un arbre binaire $\{0, 1\}$ -équilibré produit un arbre non $\{0, 1\}$ -équilibré. Ceci implique que tout couple d'éléments de $\mathcal{E}^{\{0,1\}}$ est incomparable.

Des expérimentations suggèrent également que pour tout $\beta \in \mathbb{Z}$, tout couple d'éléments de $\mathcal{E}^{\{0,\beta\}}$ est incomparable. Les ensembles $\mathcal{E}^{\{0,\beta\}}$ semblent ainsi clos par intervalle dans le treillis de Tamari, mais nous n'avons pas démontré cette propriété.

Arbres binaires $[-\alpha, \beta]$ -équilibrés

Lemme 8.4.5. *Pour tout $\alpha \geq 2$, les ensembles $\mathcal{E}^{[-\alpha, 0]}$ et $\mathcal{E}^{[-\infty, 0]}$ ne sont pas clos par intervalle dans le treillis de Tamari.*

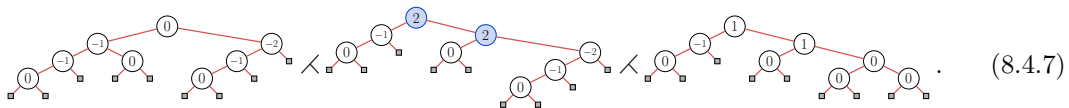
Démonstration. Il suffit d'exhiber une chaîne de la forme $T_0 \prec T_1 \prec T_2$ où $T_0, T_2 \in \mathcal{E}^{[-\alpha, 0]} \cap \mathcal{E}^{[-\infty, 0]}$ et $T_1 \notin \mathcal{E}^{[-\alpha, 0]} \cup \mathcal{E}^{[-\infty, 0]}$. La chaîne suivante, où les nœuds sont étiquetés par leurs mesures de déséquilibre, répond à ces conditions :



\square

Lemme 8.4.6. *Pour tout $\alpha \geq 2$, les ensembles $\mathcal{E}^{[-\alpha, 1]}$ et $\mathcal{E}^{[-\infty, 1]}$ ne sont pas clos par intervalle dans le treillis de Tamari.*

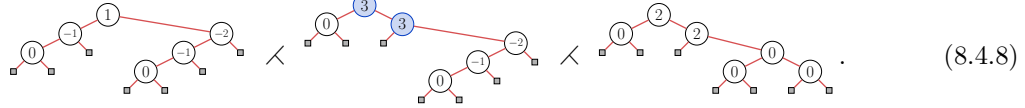
Démonstration. Il suffit d'exhiber une chaîne de la forme $T_0 \prec T_1 \prec T_2$ où $T_0, T_2 \in \mathcal{E}^{[-\alpha, 1]} \cap \mathcal{E}^{[-\infty, 1]}$ et $T_1 \notin \mathcal{E}^{[-\alpha, 1]} \cup \mathcal{E}^{[-\infty, 1]}$. La chaîne suivante, où les nœuds sont étiquetés par leurs mesures de déséquilibre, répond à ces conditions :



\square

Lemme 8.4.7. *Pour tout $\alpha \geq 2$, les ensembles $\mathcal{E}^{[-\alpha, 2]}$ et $\mathcal{E}^{[-\infty, 2]}$ ne sont pas clos par intervalle dans le treillis de Tamari.*

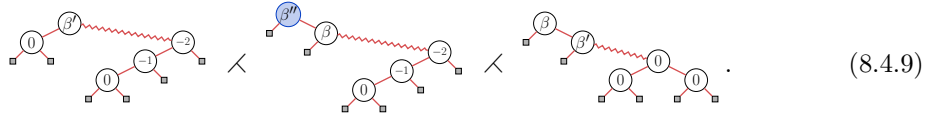
Démonstration. Il suffit d'exhiber une chaîne de la forme $T_0 \prec T_1 \prec T_2$ où $T_0, T_2 \in \mathcal{E}^{[-\alpha, 2]} \cap \mathcal{E}^{[-\infty, 2]}$ et $T_1 \notin \mathcal{E}^{[-\alpha, 2]} \cup \mathcal{E}^{[-\infty, 2]}$. La chaîne suivante, où les nœuds sont étiquetés par leurs mesures de déséquilibre, répond à ces conditions :



□

Lemme 8.4.8. Pour tous $\alpha \geq 2$ et $\beta \geq 3$, les ensembles $\mathcal{E}^{[-\alpha, \beta]}$ et $\mathcal{E}^{[-\infty, \beta]}$ ne sont pas clos par intervalle dans le treillis de Tamari.

Démonstration. Il suffit d'exhiber une chaîne de la forme $T_0 \prec T_1 \prec T_2$ où $T_0, T_2 \in \mathcal{E}^{[-\alpha, \beta]} \cap \mathcal{E}^{[-\infty, \beta]}$ et $T_1 \notin \mathcal{E}^{[-\alpha, \beta]} \cup \mathcal{E}^{[-\infty, \beta]}$. En posant $\beta' := \beta - 1$ et $\beta'' := \beta + 1$, la chaîne générique suivante, où leurs nœuds sont étiquetés par leurs mesures de déséquilibre et où les arêtes dessinées par représentent un peigne droit de taille $\beta - 3$, répond à ces conditions :



□

Théorème 8.4.9. Soit V un intervalle de \mathbb{Z} contenant 0. L'ensemble \mathcal{E}^V est clos par intervalle dans le treillis de Tamari si et seulement si $V \in \{\{0\}, \{-1, 0\}, \{0, 1\}, \{-1, 0, 1\}, \mathbb{Z}\}$.

Démonstration. Comme $\mathcal{E}^{\{0\}}$ contient uniquement des arbres binaires parfaits, et qu'il existe au plus un tel élément par ensemble \mathcal{T}_n , $\mathcal{E}^{\{0\}}$ est clos par intervalle. De plus, d'après la proposition 8.4.4, $\mathcal{E}^{\{0, 1\}}$ est clos par intervalle et par la proposition 8.4.3, $\mathcal{E}^{\{-1, 0\}}$ l'est aussi. D'après le théorème 8.1.13, $\mathcal{E}^{\{-1, 0, 1\}}$ est clos par intervalle. Finalement, comme $\mathcal{E}^{\mathbb{Z}} = \mathcal{T}$, $\mathcal{E}^{\mathbb{Z}}$ est bien entendu clos par intervalle.

Si V est un intervalle de \mathbb{Z} contenant 0 et qui ne rentre pas dans les cas considérés jusqu'à présent, alors V ou V^\sim vérifie nécessairement les hypothèses des Lemmes 8.4.5, 8.4.6, 8.4.7, ou 8.4.8. Ainsi, par la proposition 8.4.3, \mathcal{E}^V n'est pas clos par intervalle. □

Le théorème 8.4.9 souligne le rôle particulier des arbres binaires équilibrés dans le treillis de Tamari. En effet, l'intervalle $V := [-1, 1]$ de \mathbb{Z} est le seul intervalle différent de \mathbb{Z} où \mathcal{E}^V est clos par intervalle dans le treillis de Tamari et tel que le sous-poset du treillis de Tamari restreint aux éléments de \mathcal{E}^V contient des intervalles non triviaux (voir à ce propos le théorème 8.3.3 et la figure 8.9).

8.4.2 Arbres binaires équilibrés en nombre de nœuds

Rappelons que la taille $|T|$ d'un arbre binaire T est le nombre de nœuds dont il est constitué. Définissons l'application dt_T qui associe à chaque nœud x de T un élément de \mathbb{Z} , sa *mesure de déséquilibre en taille*, définie par

$$\text{dt}_T(x) := |R| - |L|, \quad (8.4.10)$$

où L (resp. R) est le sous-arbre gauche (resp. droit) de x . Un nœud x est *équilibré en taille* si

$$\text{dt}_T(x) \in \{-1, 0, 1\}. \quad (8.4.11)$$

Un arbre binaire est *équilibré en taille* si tous ses nœuds sont équilibrés en taille.

La suite $(w_n)_{n \geq 0}$, où w_n est le nombre d'arbres binaires de taille n équilibrés en taille, vérifie la relation de récurrence suivante :

$$w_n = \begin{cases} 1 & \text{si } n \in \{0, 1\}, \\ 2w_k w_{k-1} & \text{si } n = 2k, \\ w_k^2 & \text{sinon, où } n = 2k + 1. \end{cases} \quad (8.4.12)$$

C'est la suite [A110316](#) de [Slo] et ses premiers termes sont

$$1, 1, 2, 1, 4, 4, 1, 8, 16, 32, 16, 32, 16, 8, 1, 16, 64, 256, 1024, 1024. \quad (8.4.13)$$

Lemme 8.4.10. *Soit T un arbre binaire non vide équilibré en taille. Alors, la hauteur et la taille de T vérifient la relation suivante :*

$$h(T) = \lfloor \log_2(|T|) \rfloor + 1. \quad (8.4.14)$$

Démonstration. Nous procédons par induction structurelle sur l'ensemble des arbres binaires non vides équilibrés en taille. Le lemme est vrai pour l'arbre binaire à un nœud. Supposons maintenant que (8.4.14) est satisfait pour deux arbres binaires équilibrés en taille L et R tels que $T := L \wedge R$ est équilibré en taille. Nous avons deux cas à considérer, suivant si L et R sont de même taille. Si $|L| = |R|$, posons $k := |L|$. Nous avons

$$\lfloor \log_2(|T|) \rfloor + 1 = \lfloor \log_2(2k + 1) \rfloor + 1 \quad (8.4.15)$$

$$= \lfloor \log_2(2) + \log_2(k + 1/2) \rfloor + 1 \quad (8.4.16)$$

$$= \lfloor \log_2(k + 1/2) \rfloor + 2 \quad (8.4.17)$$

$$= \lfloor \log_2(k) \rfloor + 2 \quad (8.4.18)$$

$$= h(L) + 1 \quad (8.4.19)$$

$$= h(R) + 1 \quad (8.4.20)$$

$$= h(T). \quad (8.4.21)$$

L'égalité entre (8.4.17) et (8.4.18) provient du fait que k est un entier. L'égalité entre (8.4.18) et (8.4.19) suit par hypothèse d'induction.

Si $|L| \neq |R|$, supposons sans perte de généralité que $|L| = |R| + 1$ et posons $k := |L|$. Un calcul analogue au précédent implique (8.4.14). \square

Dorénavant, pour distinguer la notion d'équilibrage classique de celle d'équilibrage en taille, nous dirons qu'un arbre binaire est *équilibré en hauteur* s'il est équilibré au sens de la définition 8.1.1.

Proposition 8.4.11. *L'ensemble des arbres binaires équilibrés en taille est un sous-ensemble de l'ensemble des arbres équilibrés en hauteur.*

Démonstration. Nous procédons par induction structurelle sur l'ensemble des arbres binaires équilibrés en taille pour montrer qu'un arbre binaire équilibré en taille est aussi équilibré en hauteur. Cette propriété est vraie pour l'arbre vide et l'arbre binaire de taille un. Supposons maintenant que cette propriété est vraie pour deux arbres binaires équilibrés en taille L et R tels que $T := L \wedge R$ est équilibré en taille. D'après le lemme 8.4.10, nous avons

$$h(R) - h(L) = \lfloor \log_2(|R|) \rfloor - \lfloor \log_2(|L|) \rfloor, \quad (8.4.22)$$

et puisque T est équilibré en taille, nous avons $||R| - |L|| \leq 1$, de sorte que $|h(R) - h(L)| \leq 1$. Par hypothèse d'induction, L et R sont équilibrés en hauteur, montrant que T l'est aussi. \square

Proposition 8.4.12. *Soient T_0 et T_1 deux arbres binaires équilibrés en taille tels que $T_0 \leq_T T_1$. Alors, l'intervalle $[T_0, T_1]$ du treillis de Tamari contient uniquement des arbres binaires équilibrés en taille.*

Démonstration. Montrons que toute rotation réalisée dans un arbre binaire T augmente les mesures de déséquilibre en taille de chacun de ses nœuds. Soit y un nœud de T et x son fils gauche. Soit $(A \wedge B) \wedge C$ le sous-arbre de T de racine y et T' l'arbre binaire obtenu par la rotation en y dans T . Nous avons dans T et T' les mesures de déséquilibre en taille suivantes :

$$\begin{cases} dt_T(x) = |B| - |A|, \\ dt_T(y) = |C| - |B| - |A| - 1, \end{cases} \quad (8.4.23)$$

et

$$\begin{cases} dt_{T'}(x) = |B| + |C| + 1 - |A|, \\ dt_{T'}(y) = |C| - |B|, \end{cases} \quad (8.4.24)$$

montrant $dt_{T'}(x) > dt_T(x)$ et $dt_{T'}(y) > dt_T(y)$. Notons en outre que la rotation ne modifie par les mesures de déséquilibre en taille des autres nœuds de T .

Ceci montre que l'ensemble des arbres binaires équilibrés en taille est clos par intervalle dans le treillis de Tamari. En effet, en partant d'un arbre binaire T équilibré en taille et en réalisant une rotation produisant un arbre T' déséquilibré en taille, il existe un nœud z de T' vérifiant $dt_{T'}(z) \geq 2$ et il est impossible de décrémenter cette valeur par des rotations. Ainsi, tout arbre binaire supérieur à T' pour l'ordre de Tamari n'est pas équilibré en taille. \square

La démonstration de la proposition 8.4.12 montre également que pour tout $k \geq 0$, les ensembles des *arbres binaires k -équilibrés en taille* — qui sont les ensembles qui contiennent les arbres binaires T tels que tout nœud x de T vérifie $|dt_T(x)| \leq k$ — sont clos par intervalle dans le treillis de Tamari.

Étant donné que d'après la proposition 8.4.11, tout arbre binaire équilibré en taille est aussi équilibré en hauteur, d'après la proposition 8.4.12 et le théorème 8.3.3, les intervalles d'arbres binaires équilibrés en taille sont isomorphes à des hypercubes. Cependant, les ensembles d'arbres binaires équilibrés en taille possèdent une propriété supplémentaire :

Proposition 8.4.13. *Le sous-poset du treillis de Tamari restreint aux arbres binaires équilibrés en taille est gradué.*

Démonstration. Caractérisons les rotations qui conservent l'équilibre en taille. Soient $T_0 := (A \wedge B) \wedge C$ et $T_1 := A \wedge (B \wedge C)$ deux arbres binaires équilibrés en taille tels que $T_0 \prec_y T_1$ où y est la racine de T_0 . Notons x le fils gauche de y dans T_0 . Cette rotation ne peut pas conserver l'équilibre en taille si $dt_{T_0}(x) = 1$ ou $dt_{T_0}(y) = 1$ puisque, d'après la démonstration de la proposition 8.4.12, les mesures de déséquilibre en taille de x et de y augmentent strictement après la rotation. Voici la liste des mesures de déséquilibre en taille de x et de y dans T_0 et T_1 , exprimées sous la forme $(dt_{T_0}(x), dt_{T_0}(y)) \longrightarrow (dt_{T_1}(x), dt_{T_1}(y))$:

$$(R'1) \quad (-1, -1) \longrightarrow (2|A| - 1, |A|), \quad (R'3) \quad (-1, 0) \longrightarrow (2|A|, |A| + 1),$$

$$(R'2) \quad (0, -1) \longrightarrow (2|A| + 1, |A|), \quad (R'4) \quad (0, 0) \longrightarrow (2|A| + 2, |A| + 1).$$

Nous avons ainsi quatre cas de rotations à explorer.

Cas 1. En ce qui concerne (R'1), on a forcément $|A| = 1$. En effet, si $|A| \geq 2$, y ne serait pas équilibré en taille dans T_1 et si $|A| = 0$, comme $dt_{T_0}(x) = -1$, ceci impliquerait $|B| = -1$, ce qui

est absurde. Et donc, comme $|A| = 1$, on a $|B| = 0$ et $|C| = 1$, et le seul couple d'arbres binaires (T_0, T_1) qui convient est

$$T_0 = \begin{array}{c} \text{blue node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \\ / \quad \backslash \quad / \quad \backslash \\ \text{red node} \quad \text{red node} \quad \text{red node} \quad \text{red node} \end{array} \prec_y \begin{array}{c} \text{blue node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \\ / \quad \backslash \quad / \quad \backslash \\ \text{red node} \quad \text{red node} \quad \text{red node} \quad \text{red node} \end{array} = T_1. \quad (8.4.25)$$

Cas 2. Dans le cas (R'2), $|A| = 0$ est forcé. En effet, si $|A| \geq 1$, x ne serait pas équilibré en taille dans T_1 . Par conséquent, comme $|A| = 0$, on a $|B| = 0$ et $|C| = 0$. Le seul couple (S_0, S_1) d'arbres binaires qui convient est

$$S_0 = \begin{array}{c} \text{blue node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \end{array} \prec_y \begin{array}{c} \text{blue node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \\ / \quad \backslash \\ \text{red node} \quad \text{red node} \end{array} = S_1. \quad (8.4.26)$$

Cas 3. Dans le cas (R'3), $|A| = 0$ est forcé. Ceci implique $|B| = -1$, ce qui est absurde. Par conséquent, (R'3) ne correspond pas à une rotation qui conserve l'équilibre en taille.

Cas 4. En ce qui concerne (R'4), x vérifie $\text{dt}_{T_1}(x) \geq 2$, et (R'4) ne correspond donc pas à une rotation qui conserve l'équilibre en taille.

Il existe donc deux types de rotations qui conservent l'équilibre en taille. Ce sont celles représentées en (8.4.25) et (8.4.26).

Étant donné que chaque rotation de l'un de ces deux types supprime exactement un sous-arbre de la forme S_0 et ajoute exactement un sous-arbre de la forme S_1 , nous pouvons définir une application $\kappa : \mathcal{T} \rightarrow \mathbb{N}$ où $\kappa(T)$ est le nombre de sous-arbres de la forme S_1 que contient T . Ainsi, puisque par la proposition 8.4.12, les relations de couverture dans le treillis de Tamari restreint aux arbres binaires équilibrés en taille sont uniquement des rotations qui conservent l'équilibre en taille, la statistique κ est une fonction de graduation dans ce poset, ce qui montre qu'il est gradué. \square

8.4.3 Arbres binaires ayant une canopée fixée

Pour tout mot u sur l'alphabet $\{0, 1\}$, nous notons \mathcal{C}_u l'ensemble des *arbres binaires à canopée fixée à u* , c'est-à-dire

$$\mathcal{C}_u := \{T \in \mathcal{T} : \text{cnp}(T) = u\}. \quad (8.4.27)$$

Proposition 8.4.14. *Pour tout $u \in \{0, 1\}^*$, l'ensemble \mathcal{C}_u est clos par intervalle dans le treillis de Tamari.*

Démonstration. Commençons par montrer que \mathcal{C}_u est clos par intervalle dans le treillis de Tamari. Soit T_0 un arbre binaire et y l'un de ses nœuds. Soit $(A \wedge B) \wedge C$ le sous-arbre de T_0 de racine y et T_1 l'arbre binaire obtenu depuis T_0 par une rotation en y . Indépendamment de A et C , si B est non vide, on a $\text{cnp}(T_0) = \text{cnp}(T_1)$. Si B est vide, alors B est une feuille, et son orientation change de la droite vers la gauche. Par conséquent, $\text{cnp}(T_1)$ est plus grand pour l'ordre lexicographique que $\text{cnp}(T_0)$, ce qui montre que \mathcal{C}_u est clos par intervalle.

Donnons maintenant un argument de dénombrement pour montrer que \mathcal{C}_u est aussi un intervalle du treillis de Tamari. Soit T un maximum de \mathcal{C}_u . Toute rotation en T modifie donc sa canopée, et par conséquent, pour tout nœud y qui possède un fils gauche x dans T , x n'a pas de fils droit. L'ensemble de ces maximums, noté \mathcal{M} est caractérisé par la spécification régulière suivante (voir [FS09] pour une introduction générale sur les spécifications régulières) :

$$\mathcal{M} = \mathcal{L} \times \{\text{blue node}\} \times \mathcal{M} + \{\perp\}, \quad (8.4.28)$$

où \mathcal{L} est l'ensemble des peignes gauches. L'ensemble des maximums admet donc la série génératrice $M(t)$, qui dénombre les éléments de \mathcal{M} suivant leur nombre de nœuds vérifiant

$$M(t) = \frac{1-t}{1-2t} = 1 + \sum_{n \geq 1} 2^{n-1} t^n. \quad (8.4.29)$$

De plus, pour tout $n \geq 1$ il y a exactement 2^{n-1} ensembles \mathcal{C}_u où $|u| = n - 1$, et il y a le même nombre de ces arbres maximaux. Ceci implique qu'il existe exactement un maximum dans \mathcal{C}_u , qui est donc son plus grand élément. Le même raisonnement montre également qu'il existe exactement un plus petit élément dans chaque ensemble \mathcal{C}_u , ce qui implique le résultat. \square

Le résultat établi par la proposition 8.4.14 est déjà connu [LR02]. Seule la démonstration que nous proposons est originale.

8.4.4 Arbres binaires ayant un indice de Narayana fixé

L'indice de Narayana $\text{nar}(T)$ d'un arbre binaire T est le nombre de ses nœuds qui possèdent un fils droit non vide. Chaque ensemble d'arbres binaires de taille \mathcal{T}_n peut ainsi être partitionné en sous-ensembles $\text{Nar}(n, k) := \{T \in \mathcal{T}_n : \text{nar}(T) = k\}$ pour $0 \leq k \leq n - 1$ d'arbres binaires ayant un indice de Narayana fixé. Les nombres de Narayana [Nar55] dénombrent ces ensembles selon les deux paramètres n et k (voir la suite A001263 de [Slo]). Les premières valeurs sont

n	# Nar(n, k) pour $k = 0, \dots, n-1$							
1	1							
2	1	1						
3	1	3	1					
4	1	6	6	1				
5	1	10	20	10	1			
6	1	15	50	50	15	1		
7	1	21	105	175	105	21	1	
8	1	28	196	490	490	196	28	1

Proposition 8.4.15. *Pour tous arbres binaires T et T' tels que $\text{nar}(T) = \text{nar}(T')$ et $T \leq_{\mathcal{T}} T'$, l'intervalle $[T, T']$ du treillis de Tamari contient uniquement des éléments T'' qui vérifient $\text{nar}(T'') = \text{nar}(T) = \text{nar}(T')$.*

Démonstration. Soient T_0 et T_1 deux arbres binaires tels que $T_0 \prec_y T_1$ où y est un nœud de T_0 et $(A \wedge B) \wedge C$ le sous-arbre de T_0 de racine y . Indépendamment de A et de C , si B est non vide, T_0 et T_1 possèdent le même nombre de nœuds qui ont un fils droit non vide. Si B est vide, le nombre de fils droits est incrémenté dans T_1 . Par conséquent, dans toute chaîne $T_0 \leq_{\mathcal{T}} T_1 \leq_{\mathcal{T}} \dots \leq_{\mathcal{T}} T_\ell$, nous avons $\text{nar}(T_0) \leq \text{nar}(T_1) \leq \dots \leq \text{nar}(T_\ell)$. Ceci montre que l'ensemble des arbres binaires qui possèdent un indice de Narayana fixé est clos par intervalle dans le treillis de Tamari. \square

Proposition 8.4.16. *Pour tous $n \geq 0$ et $0 \leq k \leq n - 1$, l'ensemble $\text{Nar}(n, k)$ est une union disjointe des ensembles \mathcal{C}_u où $|u| = n - 1$ et u contient k occurrences de 1.*

Démonstration. Il est suffisant de montrer que pour tout arbre binaire T de canopée u , le nombre de 1 dans u est $\text{nar}(u)$. Montrons cette propriété par induction structurelle sur l'ensemble des arbres binaires. Si T est l'arbre vide, la propriété est clairement vraie. Supposons maintenant $T := L \wedge R$, et posons $v := \text{cnp}(L)$ et $w := \text{cnp}(R)$. Nous avons quatre cas à examiner suivant si L et R sont vides.

Cas 1. Si L et R sont vides, T est l'arbre binaire de taille un et la propriété est vérifiée.

Cas 2. Si L et R sont tous deux non vides, alors $\text{cnp}(T) = v \cdot 0 \cdot 1 \cdot w$. Puisque $\text{nar}(T) = \text{nar}(L) + \text{nar}(R) + 1$, par hypothèse d'induction, la propriété est vraie.

Cas 3. Si L est vide mais pas R , alors $\text{cnp}(T) = 1 \cdot w$. Puisque $\text{nar}(T) = \text{nar}(R) + 1$, par hypothèse d'induction, la propriété est vérifiée.

Cas 4. Si R est vide mais L , alors $\text{cnp}(T) = v \cdot 0$. Puisque $\text{nar}(T) = \text{nar}(L)$, par hypothèse d'induction, la propriété est satisfaite. \square

Corollaire 8.4.17. *Pour tous $n \geq 0$ et $0 \leq k \leq n - 1$, l'ensemble $\text{Nar}(n, k)$ est une union disjointe d'intervalles du treillis de Tamari de la forme $[T, T']$ où T (resp. T') est un élément minimal (resp. maximal) de $\text{Nar}(n, k)$.*

Démonstration. D'après la proposition 8.4.16, $\text{Nar}(n, k)$ est union disjointe d'ensembles \mathcal{C}_u , et d'après la proposition 8.4.14, ces ensembles sont des intervalles du treillis de Tamari. Le résultat s'ensuit de la proposition 8.4.15 qui montre que les plus grands éléments des \mathcal{C}_u sont les maximum de $\text{Nar}(n, k)$. \square

Perspectives

Le travail réalisé pendant cette thèse nous ouvre de nombreuses pistes de recherche. Nous faisons ici le point sur les principales questions qui restent à explorer et que nous souhaitons approfondir à moyen et à long terme.

Le monoïde de Baxter

Le chapitre 4 était consacré à l'introduction et à l'étude du monoïde de Baxter. Comme nous l'avons montré, ce monoïde possède de nombreuses similarités avec le monoïde plaxique et le monoïde sylvestre. Nous avons ainsi construit entre autres un analogue de l'algorithme d'insertion et de la correspondance de Robinson-Schensted dans le contexte du monoïde de Baxter. Il subsiste cependant sur ce monoïde de nombreuses questions. Par exemple, il serait intéressant d'avoir une formule qui, sur la donnée d'un couple d'arbres binaires de recherche jumeaux, permette de calculer le cardinal de la classe de Baxter que ce dernier encode. Par ailleurs, nous avons montré que les éléments d'une classe de Baxter sont exactement les extensions linéaires d'un poset que l'on obtient directement à partir d'un couple d'arbres binaires de recherche jumeaux. La question de caractériser ces posets se pose.

Nous avons vu que la relation d'équivalence de Baxter est également une congruence du permuttoèdre. De ce fait, celle-ci mène à la construction du treillis de Baxter, un treillis — défini initialement par Reading — sur l'ensemble des couples d'arbres binaires jumeaux. Hormis une description de ses relations de couverture et le moyen que nous avons exposé pour comparer deux couples d'arbres binaires jumeaux, nous savons assez peu de choses sur ce treillis. Une question immédiate comme le dénombrement de ses intervalles semble digne d'intérêt, d'autant plus que sa similarité avec le treillis de Tamari suggère l'utilisation de techniques analogues à celles mises en œuvre par Chapoton pour dénombrer les intervalles du treillis de Tamari.

Sous-algèbres de Hopf de \mathbf{FQSym}

Le début du chapitre 5 était consacré à rappeler la construction qui à un bon monoïde M associe une sous-algèbre de Hopf $M\mathbf{FQSym}$ de \mathbf{FQSym} . Nous avons ensuite déterminé des conditions suffisantes sur M pour que $M\mathbf{FQSym}$ vérifie diverses propriétés algébriques : pour assurer qu'elle soit autoduale, que le produit de deux éléments de sa base fondamentale s'exprime comme un intervalle d'un treillis quotient du permuttoèdre et que sa base fondamentale soit une base booléenne. Nous donnons également des conditions suffisantes sur M pour construire des bases multiplicatives et obtenir des bases de générateurs algébriques. Il serait souhaitable d'obtenir des conditions nécessaires et suffisantes pour toutes ces propriétés. De plus, lorsque $M\mathbf{FQSym}$ est autoduale, nous souhaiterions exprimer explicitement un isomorphisme entre $M\mathbf{FQSym}$ et son

dual. Ceci permettrait entre autres de construire un isomorphisme entre **Baxter** et **Baxter**^{*}, qui, comme nous l'avons montré, sont des algèbres de Hopf isomorphes.

Construction d'opérades à partir de monoïdes

Nous avons défini dans le chapitre 6 le foncteur T qui permet de construire une opérade ensembliste à partir d'un monoïde. En utilisant ce foncteur, nous avons construit plusieurs nouvelles opérades ns ensemblistes sur divers objets combinatoires. Certaines d'entre elles, comme $FCat^{(k)}$, $Schr$, $Motz$, $Comp$ et $SComp$ semblent être quadratiques. Des démonstrations de ces propriétés, ainsi que la question de savoir si ces opérades sont de Koszul, sont des suites naturelles au travail réalisé. Il reste conjointement de nombreuses opérades à construire ; par exemple, des expérimentations montrent que la sous-opérade ns de $T\{a, b\}^*$ engendrée par $\epsilon\epsilon$, ϵa et ba est indexée par l'ensemble des plantes non croisées. Parallèlement, s'il nous semble relativement facile de comprendre la structure des algèbres sur l'opérade TM où M est un monoïde, il nous paraît intéressant et plus difficile de comprendre la structure des algèbres sur les opérades \mathcal{P} où \mathcal{P} est une sous-opérade de TM .

Le foncteur T , tel qu'il est défini pour le moment, permet de construire uniquement des opérades ensemblistes. Il nous semble possible d'étendre ce foncteur de sorte qu'il prenne en entrée, au lieu d'un monoïde, une algèbre associative et retourne, au lieu d'une opérade ensembliste, une opérade dans la catégorie des espaces vectoriels. Ceci généraliserait le foncteur T et permettrait de construire d'autres opérades. Nous souhaitons formaliser cette construction et l'appliquer à diverses algèbres pour retrouver éventuellement des opérades déjà existantes ou en construire de nouvelles.

Réalisations polynomiales et le foncteur T

En utilisant le procédé que nous avons rappelé dans le chapitre 3 qui à une opérade ensembliste associe une algèbre de Hopf non commutative, nous avons construit l'algèbre de Hopf **APE** associée à APE , l'une des opérades que l'on obtient selon notre construction. Dans le but de fournir une réalisation polynomiale de cette algèbre de Hopf, nous avons interprété ses éléments en termes de forêts ordonnées d'arbres plans enracinés ; le produit et le coproduit s'interprètent alors directement sur ces objets. Une question est de savoir s'il est possible d'obtenir automatiquement cette réalisation polynomiale en raisonnant uniquement sur les éléments de **APE**, *i.e.*, des mots d'éléments de APE . Plus généralement, on peut chercher un procédé générique pour obtenir des réalisations polynomiales d'algèbres de Hopf obtenues à partir de sous-opérades d'opérades obtenues par le foncteur T .

Grammaires synchrones et motifs de déséquilibre

Les grammaires synchrones, introduites dans le chapitre 7, sont comme nous l'avons vu, des grammaires d'arbres avec un processus de dérivation particulier. Sous certaines conditions, elles permettent d'obtenir une équation fonctionnelle pour la série génératrice des arbres qu'elles engendrent. S'il est immédiat que toute grammaire d'arbres classique peut être simulée par une grammaire synchrone, la question de l'expressivité des grammaires synchrones se pose et mérite d'être regardée de près.

Il existe d'un autre côté une connexion qui semble à ce jour inexplorée entre les grammaires d'arbres et les opérades ensemblistes. En effet, tout ce qu'il est nécessaire de savoir faire dans une grammaire d'arbres est de substituer un arbre à un autre dans un endroit particulier de ce dernier. La structure algébrique et combinatoire qui encode le mieux la notion de substitution d'un objet dans un autre est précisément celle d'opérade ensembliste. Ainsi, en définissant de manière

adéquate une grammaire dont les éléments ne sont plus des arbres mais des éléments d'une opérade ensembliste, nous obtenons une généralisation des grammaires d'arbres. Il nous semble intéressant d'explorer cette voie pour en particulier déterminer l'expressivité des grammaires — synchrones ou non — ainsi obtenues et les séries génératrices qu'elles permettent d'écrire.

Dans le chapitre 8, nous avons introduit les motifs de déséquilibre, des motifs particuliers d'arbre binaire. Une question naturelle est de caractériser l'équivalence de Wilf sur les motifs de déséquilibre — rappelons que deux motifs sont Wilf équivalents si les éléments qui les évitent admettent la même série génératrice. Comme nous l'avons vu dans ce même chapitre, les grammaires synchrones permettent d'engendrer des arbres binaires qui évitent certains motifs de déséquilibre. L'étude de l'équivalence de Wilf sur ces motifs permettrait d'obtenir différentes grammaires synchrones, plus ou moins simples, dont les arbres engendrés seraient en bijection.

Arbres binaires équilibrés et treillis de Tamari

En étudiant le rôle des arbres binaires équilibrés dans le treillis de Tamari, nous avons obtenu plusieurs équations fonctionnelles pour des séries génératrices : celle des arbres binaires équilibrés maximaux et celle des intervalles — maximaux ou non — d'arbres binaires équilibrés dans le treillis de Tamari. L'existence de formules asymptotiques pour le dénombrement des arbres 2, 3-équilibrés et des arbres binaires équilibrés suggère l'existence de formules asymptotiques pour dénombrer les arbres binaires équilibrés maximaux et les intervalles d'arbres équilibrés.

Nous avons par ailleurs étudié divers ensembles d'arbres binaires qui sont clos par intervalle dans le treillis de Tamari. Pour compléter cette étude, il nous paraît intéressant de caractériser les ensembles V d'entiers tels que les ensembles d'arbres binaires V -équilibrés sont clos par intervalle dans le treillis de Tamari. Nous avons donné un début de réponse à cette question dans le cas particulier où V est un intervalle de \mathbb{Z} .

La question de savoir si l'ensemble des arbres binaires équilibrés forme une base d'une sous-algèbre ou d'un quotient de **PBT** reste ouverte. Malgré le rôle particulier joué par ces arbres dans le treillis de Tamari et le fait que le treillis de Tamari encode le produit de **PBT**, nous n'avons pour le moment pas découvert de telle structure. Il semble également que de nombreux points sur la combinatoire et l'algorithmique des arbres équilibrés restent à comprendre.

Implantations informatiques

Au cours de cette thèse, nous avons produit plusieurs programmes en Sage. Après un travail de réorganisation, nous souhaitons introduire dans Sage-Combinat les éléments suivants :

- diverses fonctions sur les permutations, les arbres binaires, les arbres binaires équilibrés et les couples d'arbres binaires jumeaux ;
- la construction qui associe à une relation d'équivalence \equiv le bon monoïde A^*/\equiv et divers outils pour le manipuler ;
- l'algèbre de Hopf **FQSym** et la plupart de ses bases connues ;
- la construction qui associe à un bon monoïde M l'algèbre de Hopf $MFQSym$;
- le foncteur T , ainsi que des outils pour manipuler les opérades obtenues ;
- diverses fonctions qui permettent de calculer les dimensions d'une opérade ns ensembliste binaire et quadratique sur la donnée de sa présentation ;
- les constructions qui associent à une opérade ensembliste \mathcal{P} les algèbres de Hopf $\mathcal{H}_{\mathcal{P}}$ et $\mathfrak{H}_{\mathcal{P}}$.

Bibliographie

- [Abe80] E. Abe. *Hopf algebras*. Cambridge University Press, 1980. 2, 29
- [ABP04] E. Ackerman, G. Barequet, and R. Y. Pinter. On the Number of Rectangular Partitions. *Proc. 15th ACM-SIAM Symp. on Discrete Algorithms*, 2004. 65
- [ABS03] M. Aguiar, N. Bergeron, and F. Sottile. Combinatorial Hopf algebras and generalized Dehn-Sommerville relations. *arXiv:math/0310016v1 [math.CO]*, 2003. 29
- [AL07] M. Aguiar and M. Livernet. The associative operad and the weak order on the symmetric groups. *Journal of Homotopy and Related Structures*, vol. 2(1) :57–84, 2007. 3, 49, 57
- [ANT11] J.-C. Aval, J.-C. Novelli, and J.-Y. Thibon. The $\#$ product in combinatorial Hopf algebras. *arXiv:1007.1901v2 [math.CO]*, 2011. 118
- [AS05] M. Aguiar and F. Sottile. Structure of the Malvenuto-Reutenauer Hopf algebra of permutations. *Adv. Math.*, 191 no. 2 :225–275, 2005. 4, 45, 47, 95
- [ASU86] A. Aho, R. Sethi, and J. Ullman. *Compilers : Principles, Techniques, and Tools*. Addison Wesley, 1986. 1
- [AU94] A. Aho and J. Ullman. *Foundations of Computer Science*. W. H. Freeman, 1994. 1, 5, 65, 69, 171
- [Aut94] J.-M. Autebert. *Théorie des langages et des automates*. Masson, 1994. 2, 159
- [AV10] J.-C. Aval and X. Viennot. The product of trees in the Loday-Ronco algebra through Catalan alternative tableaux. *Sem. Lothar. Combin.*, 63, 2010. 118
- [AVL62] G.M. Adelson-Velsky and E. M. Landis. An algorithm for the organization of information. *Soviet Mathematics Doklady*, 3 :1259–1263, 1962. 1, 6, 171
- [Bax64] G. Baxter. On fixed points of the composite of commuting functions. *Proceedings of the American Mathematical Society*, 15 :851–855, 1964. 7, 65, 72
- [BB09] O. Bernardi and N. Bonichon. Catalan’s intervals and realizers of triangulations. *Journal of Combinatorial Theory Series A*, 116 :55–75, 2009. 26
- [BBMF08] N. Bonichon, M. Bousquet-Mélou, and É. Fusy. Baxter permutations and plane bipolar orientations. *Electronic Notes in Discrete Mathematics*, 31 :69–74, 2008. 65
- [BF03] C. Brouder and A. Frabetti. QED Hopf algebras on planar binary trees. *J. Algebra*, no. 1 :298–322, 2003. 5, 50, 123
- [Bla10] P. Blasiak. Combinatorial Route to Algebra : The Art of Composition & Decomposition. *DMTCS*, 12 :2 :381–400, 2010. 29
- [BLL88] F. Bergeron, G. Labelle, and P. Leroux. Functional Equations for Data Structures. *Lecture Notes in Computer Science*, 294 :73–80, 1988. 10, 170

- [BLL94] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial Species and Tree-like Structures*. Cambridge University Press, 1994. 1, 10, 170
- [BM03] M. Bousquet-Mélou. Four classes of pattern-avoiding permutations under one roof : generating trees with two labels. *The electronic journal of combinatorics*, 9, 2003. 84
- [BN98] F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998. 67
- [BS81] S. N. Buris and H. P. Sankappanavar. *A Course in Universal Algebra*. Springer Verlag, 1981. 16
- [BS00] E. Babson and E. Steingrímsson. Generalized permutation patterns and a classification of the Mahonian statistic. *Séminaire Lotharingien de Combinatoire*, B44b, 2000. 7, 72, 181
- [Bur98] A. Burstein. *Enumeration of words with forbidden patterns*. PhD thesis, University of Pennsylvania, 1998. 181
- [BV73] J. M. Boardman and R. M. Vogt. *Homotopy invariant algebraic structures on topological spaces*, volume 347. Springer-Verlag, 1973. 2, 49
- [Car06] P. Cartier. *A primer of Hopf algebras*. IHES preprint, 2006. 2, 29
- [Cay57] A. Cayley. On the theory of the analytical forms called trees. *Phil. Mag.*, 13 :19–30, 1857. 1
- [CDG⁺07] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D. Lugiez, C. Löding, S. Tison, and M. Tommasi. *Tree Automata Techniques and Applications*. available online, 2007. <http://tata.gforge.inria.fr/>. 10, 159
- [Cha02] F. Chapoton. Rooted trees and an exponential-like series. *arXiv:math/0209104v1 [math.QA]*, 2002. 5, 59, 60
- [Cha04] F. Chapoton. A Hopf Operad of Forests of Binary Trees and Related Finite-Dimensional Algebras. *Journal of Algebraic Combinatorics*, 20 :311–330, 2004. 5, 50, 123
- [Cha05] F. Chapoton. On some anticyclic operads. *Algebraic & Geometric Topology*, 5 :53–69, 2005. 147
- [Cha06a] F. Chapoton. Sur le nombre d’intervalles dans les treillis de Tamari. *Séminaire Lotharingien de Combinatoire*, 55, 2006. 5, 58
- [Cha06b] F. Chapoton. The anticyclic operad of moulds. *arXiv:math/0609436v1 [math.QA]*, 2006. 5, 50, 123
- [Cha08] F. Chapoton. Operads and algebraic combinatorics of trees. *Séminaire Lotharingien de Combinatoire*, 58, Article B58c, 2008. 5, 49, 59
- [Cha10] F. Chapoton. Une opérade anticyclique sur les arbustes. *Annales mathématiques Blaise Pascal*, Tome 17, numéro 1 :17–45, 2010. 5, 50, 123
- [CHNT08] F. Chapoton, F. Hivert, J.-C. Novelli, and J.-Y. Thibon. An operational calculus for the Mould operad. *Int. Math. Res. Not.*, 9, 2008. 5
- [Cho56] N. Chomsky. Three Models for the Description of Language. *IRE Transactions on Information Theory*, 2 :113–123, 1956. 2, 159
- [CK98] A. Connes and D. Kreimer. Hopf algebras, renormalization and noncommutative geometry. *Comm. Math. Phys.*, 199 :203–242, 1998. 3, 9, 29, 40, 44
- [CL01] F. Chapoton and M. Livernet. Pre-Lie algebras and the rooted trees operad. *Internat. Math. Res. Notices*, no. 8 :395–408, 2001. 5, 50, 123
- [CL07] F. Chapoton and M. Livernet. Relating two Hopf algebras built from an operad. *International Mathematics Research Notices*, 2007. 5, 6, 58, 59

- [CLRS03] T.H. Cormen, C. E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to algorithms*. McGraw-Hill, 2003. 1, 5, 65, 69, 171
- [CS98] I. Chajda and V. Snášel. Congruences in Ordered Sets. *Mathematica Bohemica*, 123 :95–100, 1998. 8, 88
- [DG94] S. Dulucq and O. Guibert. Mots de piles, tableaux standards et permutations de Baxter. *Formal Power Series and Algebraic Combinatorics*, 1994. 7, 65, 73, 77, 82, 84
- [DHNT08] G. Duchamp, F. Hivert, J.-C. Novelli, and J.-Y. Thibon. Noncommutative symmetric functions VII : free quasi-symmetric functions revisited. *Annals of Combinatorics*, 2008. 4, 46, 47, 95, 98, 99
- [DHT02] G. Duchamp, F. Hivert, and J.-Y. Thibon. Noncommutative Symmetric Functions VI : Free Quasi-Symmetric Functions and Related Algebras. *International Journal of Algebra and Computation*, 12 :671–717, 2002. 3, 4, 29, 40, 46, 47, 95, 96, 99
- [DM47] A. Dvoretzky and Th. Motzkin. A problem of arrangements. *Duke Math. J.*, 14 :305–313, 1947. 136
- [FNT10] L. Foissy, J.-C. Novelli, and J.-Y. Thibon. Polynomial realizations of some combinatorial Hopf algebras. *arXiv:1009.2067v1 [math.CO]*, 2010. 4, 9, 29, 40, 44, 124
- [Foi02] L. Foissy. *Les algèbres de Hopf des arbres enracinés décorés*. PhD thesis, Université de Reims, 2002. 44
- [Foi07] L. Foissy. Bidendriform bialgebras, trees, and free quasi-symmetric functions. *J.Pure Appl. Algebra*, 209 :439–459, 2007. 8, 96, 107, 108, 118
- [Foi09] L. Foissy. *Algèbres de Hopf combinatoires*, 2009. Habilitation à diriger des recherches. 8, 44, 96, 106, 107
- [Fom94] S. Fomin. Dualiy of Graded Graphs. *Journal of Algebraic Combinatorics*, 3 :357–404, 1994. 112
- [Fra08] A. Frabetti. Groups of tree-expanded series. *J. Algebra*, 319, no. 1 :377–413, 2008. 5, 59
- [FS09] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009. 10, 14, 61, 169, 197
- [Ful97] W. Fulton. *Young Tableaux, with Applications to Representation Theory and Geometry*. Cambridge University Press, 1997. 68
- [GBV88] D. Gouyou-Beauchamps and X. Viennot. Equivalence of the two-dimensional directed animal problem to a one-dimensional path problem. *Advances in Applied Math.*, 9 :334–357, 1988. 143
- [Ges84] I. Gessel. Multipartite P-partitions and inner product of skew Schur functions. *Contemp. Math.*, 34 :289–301, 1984. 101
- [Gir10] S. Giraudo. Balanced binary trees in the Tamari lattice. *Formal Power Series and Algebraic Combinatorics*, pages 596–607, 2010. 10, 160, 172
- [Gir11] S. Giraudo. Algebraic and combinatorial structures on Baxter permutations. *Formal Power Series and Algebraic Combinatorics*, pages 387–398, 2011. 8, 66, 96
- [GK94] V. Ginzburg and M. Kapranov. Koszul duality for operads. *Duke Math. J.*, 76(1) :203–272, 1994. 3, 49
- [GKL⁺95] I.M. Gelfand, D. Krob, A. Lascoux, B. Leclerc, V.S. Retakh, and J.-Y. Thibon. Noncommutative symmetric functions I. *Adv. Math.*, 112, 1995. 3, 4, 95, 101
- [GR63] G. Th. Guilbaud and P. Rosenstiehl. Analyse algébrique d’un scrutin. *Mathématiques et sciences humaines*, tome 4 :9–33, 1963. 21

- [Hiv03] F. Hivert. An introduction to Combinatorial Hopf Algebras. *IOS Press*, 2003. 4, 29, 39, 42, 46, 99
- [Hiv04] F. Hivert. Combinatoire et calcul symbolique dans les algèbres de Hopf, 2004. Habilitation à diriger des recherches. 5, 66, 67, 96
- [HMU00] J. Hopcroft, R. Motwani, and J. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, 2nd edition, 2000. 2, 159
- [HN07] F. Hivert and J. Nzeutchap. Dual graded graphs in combinatorial Hopf algebras. *unpublished*, 2007. 95, 96, 97, 112
- [HNT02] F. Hivert, J.-C. Novelli, and J.-Y. Thibon. An analogue of the plactic monoid for binary search trees. *Comptes-Rendus Mathématique*, 335, Number 7 :577–580, 2002. 3, 4, 5, 29, 65, 69, 80, 95, 99, 100, 101
- [HNT03] Florent Hivert, J.-Y. Novelli, and J.-Y. Thibon. Sur quelques propriétés de l’algèbre des arbres binaires. *Comptes-Rendus de l’Académie des Sciences*, 337 :565–568, 2003. 69
- [HNT05] F. Hivert, J.-C. Novelli, and J.-Y. Thibon. The Algebra of Binary Search Trees. *Theoretical Computer Science*, 339, Issue 1 :129–165, 2005. 3, 4, 5, 65, 69, 75, 76, 79, 80, 81, 90, 95, 106, 172
- [HNT08a] F. Hivert, J.-C. Novelli, and J.-Y. Thibon. Commutative combinatorial Hopf algebras. *Journal of Algebraic Combinatorics*, 28 no. 1 :65–95, 2008. 3, 29
- [HNT08b] F. Hivert, J.-C. Novelli, and J.-Y. Thibon. Trees, functional equations, and combinatorial Hopf algebras. *European Journal of Combinatorics*, 29, Issue 7 :1682–1695, 2008. 138
- [HT72] S. Huang and D. Tamari. Problems of associativity : A simple proof for the lattice property of systems ordered by a semi-associative law. *Journal of Combinatorial Theory, Series A*, 13 :7–13, 1972. 26, 65
- [Joy81] A. Joyal. Une théorie combinatoire des séries formelles. *Advances in Mathematics*, 42 :1–82, 1981. 1
- [JR79] S. A. Joni and G.-C. Rota. Coalgebras and bialgebras in combinatorics. *Stud. Appl. Math*, 61 :93–139, 1979. 2
- [KLT97] D. Krob, B. Leclerc, and J.-Y. Thibon. Noncommutative symmetric functions II : Transformations of alphabets. *Internat. J. Alg. Comput.*, 7 :181–264, 1997. 4, 29, 101
- [KM01] M. Kapranov and Yu. Manin. Modules and Morita theorem for operads. *Amer. J. Math.*, 123(5) :811–838, 2001. 5, 59
- [Knu70] D. Knuth. Permutations, matrices, and generalized Young tableaux. *Pacific Journal of Mathematics*, 34 :709–727, 1970. 4, 65
- [Knu98] D. Knuth. *The Art of Computer Programming. Volume 3. Sorting and searching*. Addison Wesley Longman, 1998. 1, 5, 6, 10, 65, 69, 170, 171, 185
- [Knu04] D. Knuth. *The Art of Computer Programming. Volume 4, Fascicle 4. Generating all trees — History of combinatorial generation*. Addison Wesley Longman, 2004. 1, 6, 26, 92, 171
- [Kre98] D. Kreimer. On the Hopf algebra structure of perturbative quantum field theories. *Adv. Theor. Math. Phys.*, 2 :303–334, 1998. 40
- [KT97] D. Krob and J.-Y. Thibon. Noncommutative symmetric functions IV : Quantum linear groups and Hecke algebras at $q = 0$. *Journal of Algebraic Combinatorics* 6, 4 :339–376, 1997. 5, 68, 95, 101
- [KT99] D. Krob and J.-Y. Thibon. Noncommutative symmetric functions V : A degenerate version of $U_q(gl_N)$. *Internat. J. Algebra Comput.* 9, 3-4 :405–430, 1999. 68

- [Liv06] M. Livernet. A rigidity theorem for pre-Lie algebras. *J. Pure Appl. Algebra*, no. 1 :1–18, 2006. [5](#), [50](#), [58](#), [123](#)
- [Lod95] J.-L. Loday. La renaissance des opérades. *Séminaire N. Bourbaki*, exp. n°792 :47–74, 1994-1995. [49](#)
- [Lod01] J.-L. Loday. Dialgebras. *Lecture Notes in Math.*, 1763 :7–66, 2001. [5](#), [9](#), [50](#), [58](#), [105](#), [123](#), [124](#), [146](#)
- [Lod02] J.-L. Loday. Arithmetree. *J. Algebra*, 258 :275–309, 2002. [105](#)
- [Lod04] J.-L. Loday. Scindement d’associativité et algèbres de Hopf. *Séminaire et Congrès SMF*, 9 :155–172, 2004. [105](#)
- [Lod05] J.-L. Loday. Inversion of integral series enumerating planar trees. *Séminaire Lotharingien de Combinatoire*, 53, 2005. [181](#)
- [Lod08] J.-L. Loday. Generalized bialgebras and triples of operads. [arXiv:math/0611885v3 \[math.QA\]](#), 2008. [137](#)
- [Lot02] M. Lothaire. *Algebraic combinatorics on words*. Cambridge University Press, 2002. [3](#), [65](#), [68](#), [99](#)
- [LR34] D. E. Littlewood and A. R. Richardson. Group characters and algebra. *Phil. Trans. A*, 233 :99–141, 1934. [3](#), [65](#), [99](#)
- [LR98] J.-L. Loday and M. Ronco. Hopf Algebra of the Planar Binary Trees. *Advances in Mathematics*, 139 :293–309, 1998. [3](#), [4](#), [25](#), [29](#), [58](#), [65](#), [95](#), [99](#), [101](#), [106](#), [172](#)
- [LR02] J.-L. Loday and M. Ronco. Order Structure on the Algebra of Permutations and of Planar Binary Trees. *Journal of Algebraic Combinatorics*, Volume 15, Number 3 :253–270, 2002. [6](#), [25](#), [58](#), [95](#), [101](#), [106](#), [115](#), [172](#), [198](#)
- [LR03] J.-L. Loday and M. Ronco. Algèbres de Hopf colibres. *C.R. Acad. Sci. Paris, Ser. I* 337 :153–158, 2003. [137](#)
- [LR06] J.-L. Loday and M. Ronco. On the structure of cofree Hopf algebras. *J. reine angew. Math.*, 595 :123–155, 2006. [40](#)
- [LR09] J.-L. Loday and M. Ronco. Combinatorial Hopf Algebras. [arXiv:0810.0435v3 \[math.QA\]](#), 2009. [36](#)
- [LR10] S. Law and N. Reading. The Hopf algebra of diagonal rectangulations. [arXiv:1011.2118v1 \[math.CO\]](#), 2010. [8](#), [29](#), [77](#), [84](#), [88](#), [96](#), [111](#)
- [LS81] A. Lascoux and M.-P. Schützenberger. Le monoïde plaxique. *Noncommutative Structures in Algebra and Geometric Combinatorics*, pages 129–156, 1981. [3](#), [65](#), [68](#), [95](#)
- [LV10] J.-L. Loday and B. Vallette. *Algebraic operads*. à paraître, 2010. [3](#), [5](#), [49](#), [54](#), [58](#), [59](#), [62](#)
- [Mac95] I. G. Macdonald. *Symmetric Functions and Hall Polynomials*. Oxford University Press, 1995. [3](#), [46](#), [65](#), [99](#)
- [Mal93] C. Malvenuto. *Produits et coproduits des fonctions quasi-symétriques et de l’algèbre des descentes*. PhD thesis, Université du Québec À Montréal, 1993. [4](#), [41](#), [47](#), [95](#)
- [Mar06] M. Markl. Operads and PROPs. [arXiv:math/0601129v3 \[math.AT\]](#), 2006. [3](#), [49](#), [54](#)
- [May72] J. P. May. *The geometry of iterated loop spaces*, volume 271. Springer-Verlag, 1972. [2](#), [49](#)
- [MR95] C. Malvenuto and C. Reutenauer. Duality between quasi-symmetric functions and Solomon descent algebra. *J. Algebra*, 177 :967–982, 1995. [3](#), [4](#), [29](#), [40](#), [45](#), [95](#)
- [MSS02] M. Markl, S. Shnider, and J. Stasheff. *Operads in Algebra, Topology and Physics*, volume 96. Amer. Math. Soc., 2002. [49](#)
- [MY91] M. Méndez and J. Yang. Möbius species. *Adv. Math.*, 85(1) :83–128, 1991. [5](#)

- [Nar55] T.V. Narayana. Sur les treillis formés par les partitions d'un entier. *C.R. Acad. Sci. Paris Ser. I.*, 240 :1188, 1955. 198
- [Nov98] J.-C. Novelli. On the hypoplactic monoid. *Discrete Mathematics*, 217, Issues 1-3 :315–336, 1998. 5, 68, 95
- [NRT09] J.-C. Novelli, C. Reutenauer, and J.-Y. Thibon. Generalized descent patterns in permutations and associated Hopf algebras. [arXiv:0911.5615v1](https://arxiv.org/abs/0911.5615v1) [math.CO], 2009. 70, 120
- [NT04] J.-C. Novelli and J.-Y. Thibon. A Hopf algebra of parking functions. *Formal Power Series and Algebraic Combinatorics*, 2004. 3, 29
- [NT06] J.-C. Novelli and J.-Y. Thibon. Polynomial realizations of some trialgebras. *Formal Power Series and Algebraic Combinatorics*, 2006. 40
- [NT07] J.-C. Novelli and J.-Y. Thibon. Hopf algebras and dendriform structures arising from parking functions. *Fundamenta Mathematicae*, 193 :189–241, 2007. 3, 4, 29, 40, 106, 138, 145
- [NTT04] J.-C. Novelli, N. Thiéry, and J.-Y. Thibon. Hopf algebras of graphs. *Comptes-Rendus de l'Académie des Sciences, Paris Série I*, 339 :607–610, 2004. 3, 29
- [Nze08] J. Nzeutchap. *Correspondances de Schensted-Fomin algèbres de Hopf et graphes gradués en dualité*. PhD thesis, Université de Rouen, 2008. 96, 112
- [Odl82] A. M. Odlyzko. Periodic Oscillations of Coefficients of Power Series That Satisfy Functional Equations. *Advances in Mathematics*, 44 :180–205, 1982. 10, 169
- [Pal86] J. M. Pallo. Enumerating, Ranking and Unranking Binary Trees. *Computer Journal*, 29, Issue 2 :171–175, 1986. 91, 93
- [Pie91] B. C. Pierce. *Basic Category Theory for Computer Scientists*. The MIT Press, 1991. 30
- [PR95] S. Poirier and C. Reutenauer. Algèbres de Hopf de tableaux. *Ann. Sci. Math. Québec*, 19 :79–90, 1995. 4, 95, 99
- [Rea04] N. Reading. Lattice Congruences of the Weak Order. *Order*, 21, Issue 4 :315–344, 2004. 8, 77, 89
- [Rea05] N. Reading. Lattice congruences, fans and Hopf algebras. *Journal of Combinatorial Theory Series A*, 110 :237–273, 2005. 8, 84, 88, 89, 95, 96
- [Reu93] C. Reutenauer. *Free Lie algebras*. Oxford University Press, 1993. 40, 42
- [Rey06] M. Rey. A new construction of the Loday-Ronco algebra. *Formal Power Series and Algebraic Combinatorics*, 2006. 106
- [Rey07] M. Rey. Algebraic constructions on set partitions. *Formal Power Series and Algebraic Combinatorics*, 2007. 4, 5, 70, 95
- [Row10] E. S. Rowland. Pattern avoidance in binary trees. *Journal of Combinatorial Theory, Series A*, 117 :741–758, 2010. 181
- [S⁺11] W. A. Stein et al. *Sage Mathematics Software (Version 4.7.1)*. The Sage Development Team, 2011. <http://www.sagemath.org>. 6, 32, 35, 38, 54, 123
- [SCc11] The Sage-Combinat community. Sage-Combinat : enhancing Sage as a toolbox for computer exploration in algebraic combinatorics, 2011. <http://combinat.sagemath.org>. 6
- [Sch61] C. Schensted. Longest increasing and decreasing subsequences. *Canadian Journal of Mathematics*, 13 :179–191, 1961. 4, 65
- [Sch94] W. R. Schmitt. Incidence Hopf algebras. *J. Pure Appl. Algebra*, 96(3) :299–330, 1994. 5

- [Slo] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences. <http://www.research.att.com/~njas/sequences/>. 24, 42, 44, 46, 61, 68, 70, 72, 110, 129, 130, 131, 139, 140, 144, 173, 185, 195, 198
- [Sta99a] R.P. Stanley. *Enumerative Combinatorics, volume 2*. Cambridge University Press, 1999. 6, 26, 134, 171
- [Sta99b] J. Stasheff. Grafting Boardman's Cherry Trees to Quantum Field Theory. *Contemporary Mathematics*, 239, 1999. 49
- [Sta11] R.P. Stanley. *Enumerative Combinatorics, volume 1, 2nd edition*. Cambridge University Press, 2011. 3, 16
- [Swe69] M. Sweedler. *Hopf algebras*. Benjamin, 1969. 2, 29
- [Tam62] D. Tamari. The algebra of bracketings and their enumeration. *Nieuw Arch. Wisk.* (3), 10 :131–146, 1962. 6, 25, 65, 171
- [Tar87] R. E. Tarjan. *Data Structures and Network Algorithms*. Society for Industrial Mathematics, 1987. 1, 6
- [Tas06] M. Taskin. Properties of four partial orders on standard Young tableaux. *J. Comb. Theory, Ser A*, 113 :1092–1119, 2006. 99
- [vdL04] P. van der Laan. *Operads : Hopf algebras and coloured Koszul duality*. PhD thesis, Utrecht University, 2004. 5, 59
- [Vie04] X. Viennot. Up-down sequences of permutations, paths and canopy of binary trees, 2004. invited talk at the Lascouxfest, 52nd SLC, Otrrott. 25
- [Wes90] J. West. *Permutations with restricted subsequences and stack-sortable permutations*. PhD thesis, M.I.T., 1990. 1, 84
- [Wil02] H. Wilf. The patterns of permutations. *Discr. Math.*, 257 :575–583, 2002. 181
- [Zin10] G.W. Zinbiel. Encyclopedia of types of algebras 2010. [arXiv:1101.0267v1](https://arxiv.org/abs/1101.0267v1) [math.RA], 2010. 3, 49, 124, 137, 146

Index

— A —

Adelson-Velsky	171	EXTRAIREMAX	85
algèbre		EXTRAIREMIN	85
associative		INSERTIONFEUILLE	69
libre	38	ÉTIQUETAGEMIN	151
unitaire	30	alphabet	20
combinatoire	32	\prec -alphabet	39
commutative	31	ancêtre	23
connexe	31	animal dirigé	143
d'un monoïde	33	antipode	37
de Hopf	37	de \mathcal{M}	41
MFQSym	96–99	de CK	44
MFQSym [*]	98–99	application linéaire	19
Baxter	109–120	arête	23
Baxter [*]	112	arbre	23
CK	42–44	2, 3-équilibré	168
\mathcal{M}	40–42	à bourgeons	160
FQSym	44–47	binaire	14, 24–27
FSym	99	équilibré intérieur	184
APE	147–155	équilibré maximal	183
PBT	99–101	équilibré mixte	185
QSym	101	$\{0, 1\}$ -équilibré	192
Sym	101	à canopée fixée	197
combinatoire	38	croissant	26, 79
de Faa di Bruno	62	d'indice de Narayana fixé	198
des fonctions sur un groupe	60	décroissant	26, 79
de Lie	58	de Fibonacci	185
dendriforme	105	de recherche droit	26
max	106	de recherche gauche	26
de mélange	106	équilibré	173
libre	106	équilibré en taille	194
graduée	31	k -équilibré en taille	196
libre	32	parfait	168, 182
sur une opérade	56	V-équilibré	191
symétrique	32	de génération	84
tensorielle	31	de Schröder	138
algorithme		enraciné	23
EXTRAIREBAXTER	82	plan	23
		enraciné	133

- arc 23
- arité
- d'un élément d'une opérade 51
 - d'un nœud 23
- autodualité 37
- Aval 118
- B —
- base
- algébrique 32
 - booléenne 104
 - du côté gauche 36
 - multiplicative 32
- bigèbre 35
- bidendriforme 107
 - combinatoire 36
 - connexe 36
 - graduée 36
- Boardman 49
- bon monoïde 66–72
- borne
- inférieure 16
 - supérieure 16
- bourgeon 160
- branche 23
- Brouder 50
- C —
- canopée 24
- cardinaux
- des animaux dirigés 144
 - des arbres binaires 24
 - équilibrés 173
 - équilibrés en taille 194
 - équilibrés intérieurs 185
 - équilibrés maximaux 184
 - 0, 1-équilibrés 192
 - des arbres de Schröder 139
 - des arbres plans enracinés 134
 - des chemins de Motzkin 140
 - des couples d'arbres binaires jumeaux 72
 - anti-connexes 116
 - connexes 116
 - des endofonctions 129
 - des fonctions de parking 130
 - des forêts d'arbres
 - enracinés 42
 - plans enracinés 148 - des intervalles
 - d'arbres équilibrés 189
 - maximaux d'arbres équilibrés 190 - des k -chemins de Dyck 136
 - des mots tassés 131
- des partitions
- d'ensemble 70
 - d'entiers 61
- des permutations 44
- anti-connexes 46
 - connexes 46
 - de Baxter 72, 110
 - des tableaux de Young standard 68
- catégories 30
- catalyseur 167
- chaîne 16
- saturée 16
- Chajda 88
- Chapoton 50
- chemin 23
- de Motzkin 140
 - k -chemin de Dyck 136
- clôture par intervalle 16
- classe
- combinatoire 13
 - d'équivalence
 - anti-connexe 104
 - connexe 104
- coassociativité 33
- cogèbre
- coassociative counitaire 33
 - cocommutative 34
 - coconnexe 34
 - codendriforme 107
 - cograduée 34
 - colibre 34
 - combinatoire 35
 - tensorielle 34
- coidéal 34
- coloriage admissible 149
- compatibilité
- aux restrictions aux intervalles d'alphabet 67
 - avec l'involution de Schützenberger 67
 - avec la déstandardisation 66
- composante
- connexe 23
 - homogène 19
- composition d'entiers 141
- segmentées 145
- concaténation 20
- congruence
- de monoïde 66
 - de treillis 88
- Connes 44
- counité 33
- coupe admissible 43

- couple d'arbres binaires
 - de recherche jumeaux 73
 - jumeaux 72
 - anti-connexe 116
 - connexe 116
- couverture 16
- crochet de Lie 58
- cycle 23
- D —
- déconcaténation 40
- dérivation 161
- degré dans une opérade 55
- descendant 23
- descente 20
- diagramme
 - de Hasse 16
 - des arbres binaires équilibrés 188
 - de Tamari 91
 - double 92
 - ruban 141
- dimensions
 - de \mathcal{M} 41
 - de **CK** 42
 - de **FQSym** 44
 - de **APE** 148
- doublement d'alphabet 39
- dual
 - de Koszul 146
 - gradué
 - d'espace vectoriel 19
 - de bigèbre 37
- Duchamp 46
- Dulucq 73, 77, 82, 84
- E —
- élément
 - de type groupe 34
 - indécomposable 32
 - maximal d'un poset 16
 - minimal d'un poset 16
 - primitif 34
 - totalement primitif 108
- endofonction 128
- équation fonctionnelle de point fixe 166
- espace vectoriel
 - combinatoire 19
 - dual 19
 - gradué 19
 - libre sur un ensemble 18
- évaluation
 - d'un arbre à bourgeons 160
 - d'un mot 20
- extension
 - d'un poset 16
 - linéaire 16
- F —
- facteur 20
- feuille 24
- fil
 - droit 24
 - gauche 24
- Foissy 44, 107, 108, 118
- foncteur 60, 126
- fonction
 - de parking 129
 - de Schur 65, 99
 - quasi-symétrique 101
 - libre 46
 - symétrique 46
 - libre 99
 - non commutative 101
- forêt
 - d'arbres enracinés 42
 - ordonnée d'arbres plans enracinés 147
- forme linéaire 19
- Frabetti 50
- frontière 160
- G —
- générateurs algébriques 32
- Gouyou-Beauchamps 143
- grammaire synchrone 160
 - des arbres
 - 2, 3-équilibrés 168
 - binaires équilibrés 169
 - binaires équilibrés maximaux 183
 - binaires parfaits 168
 - des intervalles
 - d'arbres binaires équilibrés 188
 - maximaux d'arbres binaires équilibrés 189
- émondée 161
- localement finie 163
- non ambiguë 164
- graphe 23
 - étiqueté 23
 - acyclique 23
 - connexe 23
 - de génération 161
 - gradué en dualité 112
 - orienté 23
- greffe
 - d'arbres 23
 - binaires 25
- groupe 15

- associé à une opérade 59
- symétrique 21
- Guibert 73, 77, 82, 84
- H —
- hauteur
 - d'un arbre binaire 172
 - d'un mot 176
- Hivert 46, 67, 69, 95, 96, 99
- Huang 26
- hypercube 18, 187
- I —
- idéal
 - d'algèbre 31
 - d'augmentation 36
 - d'opérade 52
 - de bigèbre 36
 - de cogèbre 34
 - de Hopf 98
- idempotence 17
- identité de Jacobi 57
- image miroir 20
- indice de Narayana 198
- insertion
 - de Robinson-Schensted 66, 86
 - en feuille 69
 - en racine 86
- intersection de bons monoïdes 71
- intervalle 16
- invariant de déséquilibre 175
- inversion 20
- involution 21
 - de Schützenberger
 - sur les alphabets 21
 - sur les bons monoïdes 71
 - sur les mots 21
- K —
- Knuth 65
- Kreimer 44
- Krob 68
- L —
- Landis 171
- langage d'une grammaire synchrone 161
- Law 84, 96, 111
- lecture infixée 24
- Livernet 50
- Loday .. 25, 50, 58, 99, 101, 105, 106, 115, 146
- M —
- Malvenuto 45
- May 49
- mesure de déséquilibre
 - en hauteur 172
 - en taille 194
- monoïde 15
 - max 15
 - additif 15
 - associé à une opérade 59
 - de Baxter 73–93, 95
 - de Bell 70
 - des entiers naturels 15
 - des k -reculs 70
 - gradué 15
 - hypoplaxique 68
 - libre 20
 - multiplicatif 15
 - plaxique 68
 - quotient 66
 - sylvestre 68
 - #-sylvestre 69
- morphisme
 - d'algèbre 31
 - d'opérade 52
 - de bigèbre 36
 - de cogèbre 34
 - de monoïde 15
- mot 14, 20–22
 - admissible 176
 - binaire 68
 - des hauteurs 175
 - tassé 130
 - vide 20
- motif
 - de déséquilibre 182
 - de permutation 72
- N —
- noeud 23
- nombres
 - de Fibonacci 185
 - de Fuss-Catalan 136
 - de Stirling de la seconde espèce 131
- notation de Sweedler 33
- Novelli 44, 68–70, 99, 118
- Nzeutchap 95, 96
- O —
- occurrence d'un motif de déséquilibre 182
- opérade 50
 - associative 57
 - binaire 55
 - combinatoire 54
 - commutative associative 56
 - de Lie 57

- dendriforme 58, 105, 146
- des animaux dirigés 142
- des applications linéaires 55
- des arbres de Schröder 137
- des arbres plans enracinés 132
- des chemins de Motzkin 139
- des compositions d'entiers 141
 - segmentées 145
- des endofonctions 129
- des fonctions de parking 130
- des k -chemins de Dyck 135
- des mots tassés 130
- des permutations 132
- diassociative 146
- diptère 137
- dupliciale 137
- ensembliste 53
- finiment engendrée 52
- libre 55
- non symétrique 53
 - libre 55
- quadratique 55
- symétrisée 53
- opérateur de substitution
 - complète 51
 - partielle 50
- ordre
 - lexicographique 20
 - par longueur 20
 - monomial 163
- over
 - sur les arbres binaires 25
 - sur les couples d'arbres binaires jumeaux 115
 - sur les permutations 22
- P —
- \mathbb{P} -symbole
 - de Baxter 77
 - de Baxter itératif 87
 - sylvestre 80, 100
- parcours
 - en profondeur 133
 - infixe 24
- partition
 - d'ensemble 70
 - d'entiers 61
- peigne
 - droit 26
 - gauche 26
- permutation 21
 - anti-connexe 22
 - connexe 22
- de Baxter 72
 - anti-connexe 117
 - connexe 117
 - tordue 84
- permutoèdre 21
- plus grand élément 16
- plus petit élément 16
- Poirier 99
- polynôme
 - commutatif 33
 - non commutatif 33
- poset 16–18
 - gradué 16
 - quotient 88
- préfixe 20
- présentation d'une opérade 55
- produit
 - de convolution 37
 - de mélange 21
 - décalé 22
 - # 118
 - tensoriel
 - d'algèbres 31
 - d'applications linéaires 19
 - d'espaces vectoriels 19
 - de bigèbres 36
 - de cogèbres 34
- propriété de conservation 175
- Q —
- \mathbb{Q} -symbole
 - de Baxter 79
 - de Baxter itératif 87
 - sylvestre 79
- quadrangulation 88
- quotient
 - d'algèbre 31
 - d'opérade 53
 - de bigèbre 36
 - de cogèbre 34
 - de monoïde 66
- R —
- réécriture 67
- réalisation polynomiale
 - d'algèbre 38
 - d'algèbre de Hopf 40
 - de **Baxter** 111
 - de \mathcal{M} 42
 - de **FQSym** 46
 - de **FSym** 99
 - de **APE** 153
 - de cogèbre 39

- règle de Littlewood-Richardson 99
 racine 23
 Reading 77, 84, 88, 95, 111
 recul 21
 relation
 d'équivariance 51
 d'adjacence 67
 d'associativité
 en série 50
 en parallèle 50
 d'ordre totale 16
 Reutenauer 45, 70, 99
 Ronco 25, 99, 101, 106, 115
 rotation 25
 conservatrice d'équilibre 174
 en taille 197
 déséquilibrante 174
 partiellement déséquilibrante 174
 totalement déséquilibrante 174
- S —
- série
 de Hilbert
 d'un espace vectoriel 19
 d'une opérade 53
 génératrice
 d'une grammaire synchrone 165
 des arbres 2, 3-équilibrés 169
 des arbres binaires $\{0, 1\}$ -équilibrés 192
 des arbres binaires équilibrés 170
 des arbres binaires équilibrés maximaux
 183
 des arbres binaires parfaits 168
 des intervalles d'arbres binaires équi-
 brés 188
 des intervalles maximaux d'arbres binaires
 équilibrés 189
 exponentielle 13
 ordinaire 13
 Sage 32, 35, 38, 54, 123
 Schensted 65
 Snášel 88
 somme ordinale 39
 sommet 23
 source 23
 sous-
 algèbre 31
 arbre 23
 droit 24
 gauche 24
 bigèbre 36
 cogèbre 34
 monoïde 15
- mot 20
 opérade 52
 poset 16
 spécification régulière 14, 197
 standardisation 22
 substitution de permutations 50
 suffixe 20
 support 18
 symbole de Kronecker 18
- T —
- témoin de déséquilibre 174
 tableau de Young 77
 standard 68, 99
 Tamari 26
 Thibon 44, 46, 68–70, 99, 118
 transformation T 124
 transposition élémentaire 21
 treillis 17
 booléen 18
 de Baxter 90
 de Tamari 26
 des diviseurs 17
 quotient 89
- U —
- under
 sur les arbres binaires 25
 sur les couples d'arbres binaires jumeaux
 115
 sur les permutations 22
 unité 30
- V —
- Viennot 25, 118, 143
 Vogt 49
- W —
- West 84, 96

Résumé

Cette thèse se situe dans le domaine de la combinatoire algébrique et porte sur la construction de plusieurs structures combinatoires et algébriques sur différentes espèces d'arbres.

Après avoir défini un analogue du monoïde plaxique dont les classes d'équivalence sont indexées par les couples d'arbres binaires jumeaux, nous proposons un analogue de la correspondance de Robinson-Schensted dans ce contexte. À partir de ce monoïde, nous construisons une sous-algèbre de Hopf de l'algèbre de Hopf des fonctions quasi-symétriques libres dont les bases sont indexées par les couples d'arbres binaires jumeaux.

Ensuite, nous proposons un foncteur combinatoire de la catégorie des monoïdes vers la catégorie des opérades ensemblistes. En utilisant ce foncteur, nous construisons plusieurs opérades qui mettent en jeu divers objets combinatoires. Par le biais d'une construction qui à une opérade associe une algèbre de Hopf non commutative, nous obtenons à partir de l'une des opérades obtenue par notre construction, une algèbre de Hopf basée sur les forêts ordonnées d'arbres plans enracinés. Nous proposons une réalisation polynomiale de cette dernière.

Finalement, nous établissons certaines propriétés vérifiées par les arbres binaires équilibrés dans le treillis de Tamari. Nous montrons que l'ensemble des arbres binaires équilibrés y est clos par intervalle et que les intervalles d'arbres binaires équilibrés ont la forme d'hypercubes. Dans l'objectif de dénombrer ces intervalles, nous introduisons une nouvelle sorte de grammaires d'arbres, les grammaires synchrones. Celles-ci permettent d'obtenir une équation fonctionnelle de point fixe pour la série génératrice des arbres qu'elles engendrent.

Mots-clés :

combinatoire ; algorithmique ; arbre ; algèbre de Hopf ; opérade ; fonction quasi-symétrique libre ; treillis de Tamari ; correspondance de Robinson-Schensted ; réalisation polynomiale.

Abstract

This thesis comes within the scope of algebraic combinatorics and deals with the construction of several combinatorial and algebraic structures on different tree species.

After defining an analogue of the plactic monoid whose equivalence classes are indexed by pairs of twin binary trees, we propose in this context an analogue of the Robinson-Schensted correspondence. From this monoid, we construct a Hopf subalgebra of the Hopf algebra of free quasi-symmetric functions whose bases are indexed by pairs of twin binary trees.

Then, we propose a combinatorial functor from the category of monoids to the category of set-operads. Using this functor, we construct several operads that involve various combinatorial objects. Through a construction that brings a noncommutative Hopf algebra from an operad, we obtain from one of the operads obtained by our construction, a Hopf algebra based on ordered forests of planar rooted trees. We propose a polynomial realization of the latter.

Finally, we establish some properties satisfied by balanced binary trees in the Tamari lattice. We show that the set of balanced binary trees is closed by interval and that the intervals of balanced binary trees have the shape of hypercubes. To enumerate these intervals, we introduce a new kind of tree grammars, namely the synchronous grammars. They allow to obtain a fixed-point functional equation for the generating series of the generated trees.

Keywords:

combinatorics ; algorithmics ; tree ; Hopf algebra ; operad ; free quasi-symmetric function ; Tamari lattice ; Robinson-Schensted correspondence ; polynomial realization.